Google Scholar BETA

Web Images Video News Maps **more »**

mobile OR wifi OR bluetooth "collaborative filt 1800 - 2004 Search

Ad
Sc
Sc

Search only in: Business, Administration, Finance, and Economics
Engineering, Computer Science, and Mathematics
Social Sciences, Arts, and Humanities

Search in all subject areas.

Scholar **All articles** Recent articles Results **1** - **20** of about **905** for **mobile OR wifi OR bluetooth "collabor**

All Results
P Maes
J Konstan
J Herlocker
T Mitchell
J Riedl

GroupLens: applying **collaborative filtering** to Usenet news - group of 3 »
JA Konstan, BN Miller, D Maltz, JL Herlocker, LR ... - Communications of the ACM, 1997 - portal.acm.org
... High volume and personal taste makes Usenet news an ideal candidate for **collaborative filtering** techniques. R ... Applying **Collaborative Filtering** to Usenet News ...
Cited by 801 - Related Articles - Web Search - BL Direct

Peer-to-peer based recommendations for **mobile** commerce - group of 3 »
A Tveit - Proceedings of the 1st international workshop on **Mobile** ..., 2001 - portal.acm.org
... a recommender system for TV-programmes based on both **collaborative filtering** and content-based rec- ommendation strategies. Personalization for **mobile** users is ...
Cited by 34 - Related Articles - Web Search

Evaluating **collaborative filtering** recommender systems - group of 15 »
JL Herlocker, JA Konstan, LG Terveen, JT Riedl - ACM Transactions on Information Systems (TOIS), 2004 - portal.acm.org
Page 1. Evaluating **Collaborative Filtering** Recommender Systems ... 22, No. 1, January 2004. Page 3. Evaluating **Collaborative Filtering** Recommender Systems • 7 ...
Cited by 196 - Related Articles - Web Search - BL Direct

Collaborative filtering with privacy via factor analysis - group of 16 »
J Canny - Proceedings of the 25th annual international ACM SIGIR ..., 2002 - portal.acm.org
Collaborative Filtering with Privacy via Factor Analysis ... Our workhere builds on the recent paper [3] that intro- duced **collaborative filtering** with privacy. ...
Cited by 79 - Related Articles - Web Search

Clustering for **collaborative filtering** applications - group of 2 »
A Kohrs, B Merialdo - Proceedings of the International Conference on Computational ..., 1999 - citeseer.ist.psu.edu
... publish/publications.html): More Using Category-Based **Collaborative Filtering** in the ... Merialdo (1999) (Correct) Questionnaires: a Framework using **Mobile** Code for ...
Cited by 25 - Related Articles - Cached - Web Search

Agents that reduce work and information overload - group of 4 »
P Maes - Communications of the ACM, 1994 - portal.acm.org
Google, Inc. Subscribe (Full Service), Register (Limited Service, Free), Login. Search: The ACM Digital Library The Guide. ...
Cited by 1821 - Related Articles - Web Search - BL Direct

FieldWise: a **mobile** knowledge management architecture - group of 6 »
H Fagrell, K Forsberg, J Sanneblad - Proceedings of the 2000 ACM conference on Computer supported ..., 2000 - portal.acm.org

... it possible to send and receive text messages with the **mobile** phone. ... is added to the interest profile, giving the algorithm a **collaborative filtering** aspect, ie ...
Cited by 55 - Related Articles - Web Search

Measuring the Web - group of 6 »
T Bray - Proceedings of the fifth international World Wide Web ..., 1996 - portal.acm.org
... patterns for building user profiles and **collaborative filtering**, Proceedings of the ... improving Web service provision in wireless-**mobile** communications, **Mobile** ...
Cited by 212 - Related Articles - Web Search

A personalized television listings service - group of 3 »
B Smyth, P Cotter - Communications of the ACM, 2000 - portal.acm.org
... By integrating both content-based and **collaborative filtering** strategies, the ClixSmart ... as WML pages for WAP-enabled wireless devices, such as **mobile** telephones ...
Cited by 96 - Related Articles - Web Search

Agents that buy and sell - group of 2 »
P Maes, RH Guttman, AG Moukas - Communications of the ACM, 1999 - portal.acm.org
... In addition to constraint-based and **collaborative filtering** techniques, two other techniques are ... bidding, bidding via price-quantity graphs, and **mobile** agents. ...
Cited by 523 - Related Articles - Web Search - BL Direct

[BOOK] Machine Learning - group of 5 »
TM Mitchell - 1997 - McGraw-Hill Higher Education
... Yuan, Dynamic semantic location modeling in **mobile** enterprise applications ... J. Mooney ,
Ramadass Nagarajan, Content-boosted **collaborative filtering** for improved ...
Cited by 5538 - Related Articles - Web Search - Library Search

PTV: Intelligent Personalised TV Guides - group of 3 »
P Cotter, B Smyth - Proceedings of the 12th Innovative Applications of ..., 2000 - changingworlds.com
... information-filtering strategies, case-based reasoning and **collaborative filtering**, with user ... believe a similar success story will unfold as **mobile** phone users ...
Cited by 57 - Related Articles - Web Search - BL Direct

A hands-on look at Java **mobile** agents - group of 5 »
J Kiniry, D Zimmerman - Internet Computing, IEEE, 1997 - ieeexplore.ieee.org
... a profile management tool (Passport Office) and **collaborative filtering** technologies (Community ... is General Magic's initial implementation of **mobile** agents in ...
Cited by 171 - Related Articles - Web Search

[BOOK] **Mobile** Code: The Future of the Internet - group of 19 »
D Kotz, RS Gray, CHNH DARTMOUTH - 1999 - cs.dartmouth.edu
... Despite this, **mobile** devices will proliferate unchecked, since just as with ... Search engines, shopbots, portals, **collaborative filtering**, and email filtering are ...
Cited by 30 - Related Articles - View as HTML - Web Search - Library Search

MobileIQ: a framework for **mobile** information access - group of 3 »
P Chandrasekaran, A Joshi - **Mobile** Data Management, 2002. Proceedings. Third ..., 2002 - ieeexplore.ieee.org
... in the **mobile** scenario. W 3 IQ addressed this problem by using offline information gathering agents that combine content-based and **collaborative filtering** ...
Cited by 10 - Related Articles - Web Search

A scalable P2P recommender system based on distributed **collaborative filtering** - group of 2 »
P Han, B Xie, F Yang, R Shen - Expert Systems With Applications, 2004 - Elsevier
... GroupLens: an open architecture for **collaborative filtering** of netnews. ... 149–160. Tveit, A. (2001). Peer-to-peer based recommendations for **mobile** commerce. ...
Cited by 8 - Related Articles - Web Search

Mobile Virtual Healthcare Communities: An Approach To Community Engineering For Cancer Patients - group of 6 »
JM Leimeister, M Daum, H Krcmar - European Conference on Information Systems (ECIS), Gdansk, 2002 - is2.lse.ac.uk
... **Mobile** Virtual Healthcare Communities: An Approach to Community Engineering ... into four areas: Adaptability, awareness, **collaborative filtering** and interaction. ...
Cited by 16 - Related Articles - View as HTML - Web Search

Alleviating the latency and bandwidth problems in WWW browsing - group of 6 »
TS Loon, V Bharghavan - Proceedingsof the 1997 Usenix Symposium on Internet ..., 1997 - ce.sejong.ac.kr
... over slow links, and (c) temporary disconnections of **mobile** users - either ... of application adaptation in general [12, 25, 7]. **Collaborative filtering** has been ...
Cited by 54 - Related Articles - View as HTML - Web Search - Library Search

GeoNotes: social enhancement of physical space - group of 4 »
P Persson, F Espinoza, E Cacciatore - Conference on Human Factors in Computing Systems, 2001 - portal.acm.org
... terminals, people will want to do with their **mobile** devices when ... strives to socially enhance digital space (CSCW, **collaborative filtering**, social navigation). ...
Cited by 12 - Related Articles - Web Search

The Effects of Wireless Computing in Collaborative Learning Environments - group of 4 »
G Gay, M Stefanone, M Grace-Martin, H Hembrooke - International Journal of Human-Computer Interaction, 2001 - Lawrence Earlbaum
... of technology on learning, we can assume that **mobile** and wireless ... Rec- ommendation systems employ **collaborative filtering** techniques based on aggre- gate data ...
Cited by 31 - Related Articles - Web Search - BL Direct

Gooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

mobile OR wifi OR bluetooth "collab Search

Google Home - About Google - About Google Scholar

Google Scholar BETA

Web Images Video News Maps **more »**

mobile OR wifi OR bluetooth "collaborative filt 1800 - 2004 Search

Ad
Sc
Sc

Search only in: Business, Administration, Finance, and Economics
Engineering, Computer Science, and Mathematics
Social Sciences, Arts, and Humanities

Search in all subject areas.

Scholar **All articles** Recent articles Results **1** - **20** of about **492** for **mobile OR wifi OR bluetooth "collabor**

All Results
J Konstan
J Herlocker
J Riedl
G Salton
B Miller

GroupLens: applying **collaborative filtering** to Usenet news - group of 3 »
JA Konstan, BN Miller, D Maltz, JL Herlocker, LR ... - Communications of the ACM, 1997 - portal.acm.org
... 4]. • The requirement that GroupLens provide predictions of the **rating** the system ...
forward to undertake the challenge of applying **collaborative filtering** to a ...
Cited by 801 - Related Articles - Web Search - BL Direct

Peer-to-peer based recommendations for **mobile** commerce - group of 3 »
A Tveit - Proceedings of the 1st international workshop on **Mobile** ..., 2001 - portal.acm.org
... could for instance increase the **rating** of the ... TV-programmes based on both **collaborative**
filtering and content ... Personalization for **mobile** users is supported by ...
Cited by 34 - Related Articles - Web Search

Collaborative filtering with privacy via factor analysis - group of 16 »
J Canny - Proceedings of the 25th annual international ACM SIGIR ..., 2002 - portal.acm.org
... is expected value). The RMS noise for typical **collaborative filtering** domains
is about 20% of the **rating** range. The scalar ψ should ...
Cited by 79 - Related Articles - Web Search

Clustering for **collaborative filtering** applications - group of 2 »
A Kohrs, B Merialdo - Proceedings of the International Conference on Computational ..., 1999 - citeseer.ist.psu.edu
... on how to use the **rating** matrix to ... html): More Using Category-Based **Collaborative Filtering** in the ... Correct) Questionnaires: a Framework using **Mobile** Code for.. ...
Cited by 25 - Related Articles - Cached - Web Search

Evaluating **collaborative filtering** recommender systems - group of 15 »
JL Herlocker, JA Konstan, LG Terveen, JT Riedl - ACM Transactions on Information Systems (TOIS), 2004 - portal.acm.org
... Evaluating **Collaborative Filtering** Recommender Systems ... also interacts with the type of **rating**—implicit rat ... user to intentionally create a **ranking** among their ...
Cited by 196 - Related Articles - Web Search - BL Direct

Further Experiments on Collaborative **Ranking** in Community-Based Web Search - group of 2 »
J Freyne, B Smyth, M Coyle, E Balfe, P Briggs - Artificial Intelligence Review, 2004 - Springer
... retrieval tasks, for example, search on **mobile** devices such ... histories and their use
in **ranking** metrics that ... I-SPY borrows ideas from **collaborative filtering** re ...
Cited by 16 - Related Articles - Web Search - BL Direct

A personalized television listings service - group of 3 »

B Smyth, P Cotter - Communications of the ACM, 2000 - portal.acm.org
... both content-based and **collaborative filtering** strategies, the ... wireless devices, such as **mobile** telephones. ... For example, by **rating** "Ally McBeal" positively ...
Cited by 96 - Related Articles - Web Search

MobileIQ: a framework for **mobile** information access - group of 3 »
P Chandrasekaran, A Joshi - **Mobile** Data Management, 2002. Proceedings. Third ..., 2002 - ieeexplore.ieee.org
... U **rating** URL U **rating** U U D ... bandwidth limitations of the **mobile** wireless environment ...
personalization services already offered by using **collaborative filtering**. ...
Cited by 10 - Related Articles - Web Search

GeoNotes: social enhancement of physical space - group of 4 »
P Persson, F Espinoza, E Cacciatore - Conference on Human Factors in Computing Systems, 2001 - portal.acm.org
... a GeoNotes database: the top **ranking** GeoNotes will ... to do with their **mobile** devices when ... enhance digital space (CSCW, **collaborative filtering**, social navigation ...
Cited by 12 - Related Articles - Web Search

A scalable P2P recommender system based on distributed **collaborative filtering** - group of 2 »
P Han, B Xie, F Yang, R Shen - Expert Systems With Applications, 2004 - Elsevier
... where v a;j is the **rating** given to ... GroupLens: an open architecture for **collaborative filtering** of netnews ... Peer-to-peer based recommendations for **mobile** commerce ...
Cited by 8 - Related Articles - Web Search

Mobile Access to the Físchlár-News Archive - group of 2 »
C Gurrin, AF Smeaton, H Lee, K Mc Donald, N Murphy ... - ... with **Mobile** Devices and Services, Workshop on **Mobile** and ..., 2003 - Springer
... is based on news story **rating** data and ... of content similarity of news stories and **collaborative filtering**. As stated, Físchlár-News on a **mobile** device will ...
Cited by 5 - Related Articles - Web Search - BL Direct

Method for cataloging, filtering, and relevance **ranking** frame-based hierarchical information ... - group of 3 »
S Chakrabarti, BE Dom, DA Gibson, P Raghavan, S ... - US Patent 6,334,131, 2001 - Google Patents
... Publication: "Using **Collaborative Filtering** to Weave an Information ... followed by filtering and **ranking** the pages ... the subject of auto- **mobile** restoration, would ...
Cited by 7 - Related Articles - Web Search

Collaborative Filtering with Maximum Entropy - group of 11 »
D Pavlov, E Manavoglu, DM Pennock, C Lee Giles - Intelligent Systems and Their Applications, IEEE [see also ..., 2004 - ieeexplore.ieee.org
... and order- independent: that is, their **ranking** of recom ... application of maxent for **collaborative filtering** and one ... service, qos, ... 8 **mobile**, wireless, protocol ...
Cited by 5 - Related Articles - Web Search

AmbientDB: P2P data management middleware for ambient intelligence - group of 5 »
W Fontijn, P Boncz - Pervasive Computing and Communications Workshops, 2004. ..., 2004 - ieeexplore.ieee.org
... technology generally assumes that **mobile** nodes are ... Network (BAN) functionality (eg **Bluetooth**) such that ... Workshop on Filtering and **Collaborative Filtering**, 1998 ...

Cited by 10 - Related Articles - Web Search

MovieLens unplugged: Experiences with a recommender systems on four **mobile** devices - group of 3 »
BN Miller, I Albert, SK Lam, JA Konstan, J Riedl - Proceedings of the 2003 Conference on Intelligent User ..., 2003 - cs.luther.edu
... 4: Theater selection Figure 5: **Rating**, showtime ... predictive algorithms for **collaborative filtering**, in Proceedings ... MJ (2001), Improving **mobile** internet usability ...
Cited by 6 - Related Articles - View as HTML - Web Search - BL Direct

Integrating a multi-agent recommendation system into a **mobile** learning management system
A Andronico, A Carbonaro, G Casadei, L Colazzo, A ... - Artificial Intelligence in **Mobile** System, 2003 - w5.cs.uni-sb.de
... The **scoring** simplest form considers a positive score ... systems and network management,
mobile access/management ... rule mining, or **collaborative filtering**, etc [26]. ...
Cited by 5 - Related Articles - View as HTML - Web Search

PILGRIM: A location broker and mobility-aware recommendation system - group of 14 »
M Brunato, R Battiti - Pervasive Computing and Communications, 2003.(PerCom 2003). ..., 2003 - ieeexplore.ieee.org
... palmtop devices and by the mushrooming of GPRS, 3G, Wi-Fi, **Bluetooth** and other ... Location
Broker **Mobile** computer ... **Collaborative filtering** and **ranking** procedure ...
Cited by 16 - Related Articles - Web Search

Term-weighting approaches in automatic text retrieval - group of 6 »
G Salton, C Buckley - Information Processing and Management: an International ..., 1988 - portal.acm.org
... System, IEEE Transactions on **Mobile** Computing, v.1 n ... Liming Ren, Document **ranking**
on weight ... using a distributed **collaborative filtering** architecture, Proceedings ...
Cited by 1735 - Related Articles - Web Search - Library Search

A Novel Distributed **Collaborative Filtering** Algorithm and Its Implementation on P2P Overlay Network - group of 3 »
P Han, B Xie, F Yang, R Shen - Proc. of PAKDD, 2004 - Springer
... Where j a v , is the **rating** given to ... 6. Eachmovie **collaborative filtering** data set.:
http://research.compaq.com ... to-peer based Recommendations for **Mobile** Commerce ...
Cited by 2 - Related Articles - Web Search - BL Direct

User Preferences Initialization and Integration in Critique-Based **Mobile** Recommender Systems - group of 4 »
QN Nguyen, F Ricci - ... on Artificial Intelligence in **Mobile** Systems,(AIMS'04), 2004 - ectrl.itc.it
... like-minded users (in the **collaborative filtering** approach [8 ... we use a GPS device
via a **Bluetooth** connection with ... In **mobile** recommender systems, the space-time ...
Cited by 3 - Related Articles - View as HTML - Web Search

Goooooooooogle ▶

Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

mobile OR wifi OR bluetooth "collab Search

Set Items Description
S1 377165 S PDA OR PDAS OR PERSONAL()DIGITAL()ASSISTANT? ? OR HANDHELD? ? OR HAND()HELD? ? OR PALM OR PALMTOP? ? OR PALMPILOT? ? OR POCKET()PC? ? OR PSP OR BLACKBERRY? ? OR BLACKBERRIES OR LAPTOP? ? OR NOTEBOOK? ? OR THINKPAD? ? OR TRIO OR TRIOS OR TREO OR TREOS OR PIT OR PAGER? ? OR PERSONAL()DEVICE? ? OR HANDSPRING? ? OR NEWTON? ? OR PEN(3W)(COMPUTER? ? OR DEVICE? ?) OR CELLPHONE? ? OR MOBILEPHONE? ? OR SMARTPHONE? ? OR (CELL OR MOBILE OR SMART)()PHONE? ?
S2 13572360 S RANK?? OR RANKING OR GRADE? ? OR GRADING OR SCORE? ? OR SCORING OR RATE? ? OR RATED OR RATING? ? OR WEIGHT??? OR SCALE? ? OR SCALING OR EVALUATE? ? OR EVALUATING OR EVALUATION? ? OR RECOMMEND????
S3 3135 S COLLABORATIVE()FILTERING OR RECOMMENDATION()SYSTEM? ?
S4 214 S SEMANTIC()(TREE OR TREES)
S5 37 S S1 AND S2 AND S3
S6 0 S S5 AND S4
S7 20 RD S5 (unique items)
S8 1775649 S WIRELESS OR WI()FI OR WIFI OR 802()11 OR AIRPORT OR PORTABLE? ? OR MOBILE? ? OR REMOTE OR REMOTELY OR AD()HOC OR MANET OR GSM OR GROUPE()SPECIAL()MOBILE OR BLUETOOTH OR GLOBAL()SYSTEM()FOR()MOBILE()COMMUNICATION? ? OR CDMA OR 1XRTT OR EV()DO OR EVOLUTION()DATA()ONLY OR WIMAX OR WORLD()INTEROPERABILITY(2W)MICROWAVE()ACCESS OR BWA
S9 371658 S PEER(1W)PEER OR P2P OR ((GRID OR DISTRIBUTED OR UTILITY)()COMPUTING) OR MULTICOMPUTER OR MULTI()COMPUTER OR MESH()(NETWORK OR LAN) OR (PARALLEL OR DISTRIBUTED OR GRID)(1W)(PROCESS??? OR COMPUT? OR RESOURCES) OR CLUSTER?(1W)(COMPUT? OR SUPERCOMPUT?)
S10 523691 S S2 (5N) (SHARE? ? OR SHARING OR EXCHANGE? ? OR EXCHANGING OR DISTRIBUTE? ? OR DISTRIBUTING OR DISTRIBUTION? ? OR COLLECT???? OR AGGREGATE? ? OR AGGREGATING OR AGGREGATION? ? OR CONSOLIDATE? ? OR CONSOLIDATING OR CONSOLIDATION? ? OR MERGE? ? OR MERGING OR COMBINE? ? OR COMBINING OR COMBINATION? ?)
S11 1299 S (S1 OR S8) AND S9 AND S10
S12 3 S S11 AND S3
S13 2 S S12 NOT S7
S14 8 S (S1 OR S8) AND S10 AND S3
S15 4 S S14 NOT (S7 OR S13)
S16 3 RD (unique items)

? show files

[File 8] **Ei Compendex(R)** 1884-2007/Apr W3

[File 35] **Dissertation Abs Online** 1861-2007/Mar

[File 65] **Inside Conferences** 1993-2007/Apr 20

[File 2] **INSPEC** 1898-2007/Apr W3

[File 111] **TGG Natl.Newspaper Index(SM)** 1979-2007/Apr 19

[File 6] **NTIS** 1964-2007/Apr W3

[File 144] **Pascal** 1973-2007/Apr W3

[File 434] **SciSearch(R) Cited Ref Sci** 1974-1989/Dec

[File 34] **SciSearch(R) Cited Ref Sci** 1990-2007/Apr W3

[File 62] **SPIN(R)** 1975-2007/Apr W2

[File 99] **Wilson Appl. Sci & Tech Abs** 1983-2007/Mar

[File 95] **TEME-Technology & Management** 1989-2007/Apr W4

[File 56] **Computer and Information Systems Abstracts** 1966-2007/Apr

[File 57] **Electronics & Communications Abstracts** 1966-2007/Apr

[File 60] **ANTE: Abstracts in New Tech & Engineer** 1966-2007/Apr

[File 266] **FEDRIP** 2007/Mar

[File 583] **Gale Group Globalbase(TM)** 1986-2002/Dec 13

**File 583: This file is no longer updating as of 12-13-2002.*

[File 438] **Library Lit. & Info. Science** 1984-2007/Mar

[File 256] **TecInfoSource** 82-2007/Apr

7/5/5 (Item 5 from file: 8) **Links**

Fulltext available through: Institution of Electrical Engineers

10270849 **E.I. No:** EIP05088845410

Title: View through MetaLens: Usage patterns for a meta-recommendation system

Author: Schafer, J.B.; Konstan, J.A.; Riedl, J.
Corporate Source: Department of Computer Science University of Northern Iowa, Cedar Falls, IA 50614-0507, United States

Source: IEE Proceedings: Software v 151 n 6 November 2004.
Publication Year: 2004
CODEN: IPSEFU **ISSN:** 1462-5970
Language: English
Document Type: JA; (Journal Article) **Treatment:** T; (Theoretical)
Journal Announcement: 0503W1

Abstract: In a world where a person's number of choices can be overwhelming, **recommender** systems help users find and **evaluate** items of interest. They do so by connecting users with information regarding the content of **recommended** items or the opinions of other individuals. Such systems have become powerful tools in domains such as electronic commerce, digital libraries and knowledge management. The authors address such systems, as well as a relatively new class of **recommender** system called meta-**recommenders**. Meta-**recommenders** provide users with personalised control over the generation of a single recommendation list formed from a combination of rich data using multiple information sources and recommendation techniques. They discuss observations made from the public trial of a meta-**recommender** system in the domain of movies and lessons learned from the incorporation of features that allow persistent personalisation of the system. Finally, they consider the challenges of building real-world, usable meta- **recommenders** across a variety of domains. copy IEE, 2004. 30 Refs.

Descriptors: *User interfaces; Computer systems; Algorithms; Data mining; Information retrieval; **Personal digital assistants**; Internet; Electronic commerce; Statistical methods
Identifiers: Meta-**recommendation system**; MetaLens; Information filtering; **Collaborative filtering**
Classification Codes:
722.2 (Computer Peripheral Equipment); 723.2 (Data Processing); 903.3 (Information Retrieval & Use); 722.4 (Digital Computers & Systems) ; 723.5 (Computer Applications); 922.2 (Mathematical Statistics)
722 (Computer Hardware); 723 (Computer Software, Data Handling & Applications); 903 (Information Science); 922 (Statistical Methods)
72 (COMPUTERS & DATA PROCESSING); 90 (ENGINEERING, GENERAL); 92 (ENGINEERING MATHEMATICS)

7/5/6 (Item 6 from file: 8) **Links**
Fulltext available through: USPTO Full Text Retrieval Options

10215153 **E.I. No:** EIP05028782614
Title: Personalisation services for learning management systems in mobile settings

Author: Andronico, Alfio; Carbonaro, Antonella; Colazzo, Luigi; Molinari, Andrea
Corporate Source: Department of Computer Science University of Bologna, Bologna I-40127, Italy
Source: International Journal of Continuing Engineering Education and Life-Long Learning v 14 n 4-5 2004. p 353-369
Publication Year: 2004
CODEN: ICEEE4 **ISSN:** 1560-4624
Language: English
Document Type: JA; (Journal Article) **Treatment:** T; (Theoretical)
Journal Announcement: 0501W3
Abstract: The paper presents the guidelines and first findings of a project of three Italian universities (Bologna, Siena and Trento) which aim is to offer personalisation services in a mobile learning platform. The integration of a **recommendation system** that suggests educational resources to students improves the user's adaptivity and personalisation. Technological aspects of this integration are illustrated. Besides, the extensions of a Learning Management System for mobile learning are presented, including services for the mobile users of the system. The project has its foundations in the availability of significant experience on e-learning real processes, and on the availability of an e-learning system developed in previous projects and currently used by different faculties. 16 Refs.

Descriptors: *Learning systems; Societies and institutions; Project management; XML; **Personal digital assistants**; Teaching; Mobile telecommunication systems; Standards
Identifiers: Mobile learning; Personalization mechanisms; **Collaborative filtering; Recommender** systems
Classification Codes:
901.1.1 (Societies & Institutions)
723.4 (Artificial Intelligence); 901.1 (Engineering Professional Aspects) ; 912.2 (Management); 901.2 (Education); 902.2 (Codes & Standards)
723 (Computer Software, Data Handling & Applications); 901 (Engineering Profession); 912 (Industrial Engineering & Management); 722 (Computer Hardware); 716 (Electronic Equipment, Radar, Radio & Television); 718 (Telephone & Other Line Communications); 902 (Engineering Graphics; Engineering Standards; Patents)
72 (COMPUTERS & DATA PROCESSING); 90 (ENGINEERING, GENERAL); 91 (ENGINEERING MANAGEMENT); 71 (ELECTRONICS & COMMUNICATION ENGINEERING)

7/5/7 (Item 7 from file: 8) Links
Fulltext available through: custom link USPTO Full Text Retrieval Options
Ei Compendex(R)

10203319 **E.I. No:** EIP05018769064
Title: VISCORS: A visual-content recommender for the mobile Web

Author: Kim, Chan Young; Lee, Jae Kyu; Cho, Yoon Ho; Kim, Deok Hwan
Corporate Source: Dept. of Internet Business Dongyang Technical College, Kuro, Seoul, 152-714, South Korea
Source: IEEE Intelligent Systems v 19 n 6 November/December 2004. p 32-40
Publication Year: 2004
ISSN: 1541-1672
Language: English
Document Type: JA; (Journal Article) **Treatment:** G; (General Review); X; (Experimental)
Journal Announcement: 0501W2
Abstract: The advantages of VISCORS, a visual contents **recommender** system, for the mobile web are discussed. VISCORS combines the two popular information filtering techniques, **collaborative filtering** and content-based image retrieval. With the help of VISCORS, the customers can purchase content with much less search effort and much lower connection time. It also helps mobile web content providers in helping the profitability of their business because lower customer frustration in finding desired content increases revenue through an improved purchase conversion **rate**. (Edited abstract) 10 Refs.
Descriptors: *World Wide Web; Mobile computing; Wireless telecommunication systems; **Personal digital assistants**; Intelligent agents; Image retrieval; Information analysis; Web browsers; Real time systems; Database systems; Multimedia systems; Feedback
Identifiers: Mobile web; **Cell phones**; Wireless devices; **Collaborative filtering**; Kookmin University
Classification Codes:
723.4 (Artificial Intelligence); 903.1 (Information Sources & Analysis); 722.4 (Digital Computers & Systems); 723.3 (Database Systems); 723.5 (Computer Applications); 731.1 (Control Systems)
723 (Computer Software, Data Handling & Applications); 716 (Electronic Equipment, Radar, Radio & Television); 722 (Computer Hardware); 903 (Information Science); 731 (Automatic Control Principles & Applications)
72 (COMPUTERS & DATA PROCESSING); 71 (ELECTRONICS & COMMUNICATION ENGINEERING); 90 (ENGINEERING, GENERAL); 73 (CONTROL ENGINEERING)

7/5/8 (Item 8 from file: 8) Links
Fulltext available through: ACM - Association for Computing Machinery USPTO Full Text Retrieval Options
Ei Compendex(R)

09990262 **E.I. No:** EIP04348323863
Title: PocketLens: Toward a personal recommender system

Author: Miller, Bradley N.; Konstan, Joseph A.; Riedl, John
Corporate Source: Computer Science Department Luther College, Decorah, IA 52101, United States
Source: ACM Transactions on Information Systems v 22 n 3 July 2004. p 437-476
Publication Year: 2004
CODEN: ATISET **ISSN:** 1046-8188
Language: English
Document Type: JA; (Journal Article) **Treatment:** L; (Literature Review/Bibliography); T; (Theoretical)
Journal Announcement: 0408W4
Abstract: Recommander systems using **collaborative filtering** are a popular technique for reducing information overload and finding products to purchase. One limitation of current **recommenders** is that they are not portable. They can only run on large computers connected to the Internet. A second limitation is that they require the user to trust the owner of the **recommender** with personal preference data. Personal **recommenders** hold the promise of delivering high quality recommendations on **palmtop** computers, even when disconnected from the Internet. Further, they can protect the user's privacy by storing personal information locally, or by sharing it in encrypted form. In this article we present the new PocketLens **collaborative filtering** algorithm along with five peer-to-peer architectures for finding neighbors. We **evaluate** the architectures and algorithms in a series of offline experiments. These experiments show that Pocketlens can run on connected servers, on usually connected workstations, or on occasionally connected portable devices, and produce recommendations that are as good as the best published algorithms to date. 66 Refs.
Descriptors: *Computer supported cooperative work; Database systems; Internet; **Personal digital assistants**; Computer architecture; Information technology; Computer software portability; Security of data; Data privacy; Algorithms
Identifiers: Collaborative filtering; Peer-to-peer networking; **Recommender** systems; Privacy
Classification Codes:
723.5 (Computer Applications); 723.3 (Database Systems); 723.2 (Data Processing)
723 (Computer Software, Data Handling & Applications); 722 (Computer Hardware); 903 (Information Science)
72 (COMPUTERS & DATA PROCESSING); 90 (ENGINEERING, GENERAL)

7/5/9 (Item 9 from file: 8) **Links**
Ei Compendex(R)

09468937 **E.I. No:** EIP03317571028
Title: MovieLens unplugged: Experiences with an occasionally connected recommender system

Author: Miller, Bradley N.; Albert, Istvan; Lam, Shyong K.; Konstan, Joseph A.; Riedl, John
Corporate Source: GroupLens Research Group University of Minnesota, Minneapolis, MN 55455, United States
Conference Title: 2003 International Conference on Intelligent User Interfaces
Conference Location: Miami, FL, United States **Conference Date:** 20030112-20030115
Sponsor: ACM SIGART , ACM SIGCHI
E.I. Conference No.: 61240
Source: International Conference on Intelligent User Interfaces, Proceedings IUI 2003. p 263-266
Publication Year: 2003
Language: English
Document Type: CA; (Conference Article) **Treatment:** T; (Theoretical)
Journal Announcement: 0308W1
Abstract: Recommander systems have changed the way people shop online. **Recommender** systems on wireless mobile devices may have the same impact on the way people shop in stores. We present our experience with implementing a **recommender** system on a **PDA** that is occasionally connected to the network. This interface helps users of the MovieLens movie recommendation service select movies to rent, buy, or see while away from their computer. The results of a nine month field study show that although there are several challenges to overcome, mobile **recommender** systems have the potential to provide value to their users today. 13 Refs.
Descriptors: *Information science; Mobile telecommunication systems; **Personal digital assistants**; Telecommunication networks; Computer systems; User interfaces
Identifiers: Collaborative filtering
Classification Codes:
722.2 (Computer Peripheral Equipment)
903 (Information Science); 716 (Electronic Equipment, Radar, Radio & Television); 722 (Computer Hardware); 723 (Computer Software, Data Handling & Applications)
90 (ENGINEERING, GENERAL); 71 (ELECTRONICS & COMMUNICATION ENGINEERING); 72 (COMPUTERS & DATA PROCESSING)

7/5/11 (Item 1 from file: 35) **Links**
Dissertation Abs Online

01926113 ORDER NO: AADAA-I3076327
Toward a personal recommender system

Author: Miller, Bradley Norman
Degree: Ph.D.
Year: 2003
Corporate Source/Institution: University of Minnesota (0130)
Advisers: John T. Riedl; John Vincent Carlis
Source: Volume 6312B of Dissertations Abstracts International.
PAGE 5937 . 185 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984
ISBN: 0-493-96703-6

Recommender systems using **collaborative filtering** are a popular technique for reducing information overload and finding products to purchase. However the economic model required to run a business around **collaborative filtering** is at odds with the end user's de sire for unvarnished recommendations. Traditional **recommender** systems are centralized and available only online which is at odds with the user's desire to have recommendations wherever they are. A personal **recommender** system will someday empower people with the technology needed to assert their freedom to share information of all kinds, and to take recommendations with them, wherever they go.

In this thesis we take three steps toward the long term vision of a personal **recommender** system. The PocketLens peer-to-peer **collaborative filtering** algorithm, the MultiLens recommendation framework, and MovieLens Unplugged.

We present the PocketLens **collaborative filtering** algorithm along with four peer-to-peer architectures for finding neighbors. We **evaluate** the architectures and algorithms in a series of experiments. These experiments show that PocketLens can run on portable and disconnected devices, give users control of their data, and produce recommendations that are as good as the best published algorithms to date.

We present the MultiLens framework. A new recommendation engine capable of combining multiple dimensions of preference and content information into a model used to make recommendations. We identify twelve application patterns used by **recommender** applications, and show how the MultiLens framework can be used to implement these patterns. We experimentally **evaluate** the ability of MultiLens to combine a content dimension with a quality dimension to solve the first **rater** and sparsity problems in **collaborative filtering**.

We present MovieLens unplugged, which examines several important challenges that interface designers must overcome on mobile devices: Providing sufficient value to attract prospective wireless users, handling occasionally connected devices, privacy and security, and surmounting the physical limitations of the devices. We present our experience with the implementation of a wireless movie **recommender** system on a **cell phone** browser, an AvantGo channel, a wireless **PDA** , and a voice-only phone interface. These interfaces help MovieLens users select movies to rent, buy, or see while away from their computer.

Great reference

Wrong date

7/5/12 (Item 1 from file: 2) Links
INSPEC

10219046
Title: Location-based service with context data for a restaurant
Author Bae-Hee Lee; Heung-Nam Kim; Jin-Guk Jung; Geun-Sik Jo
Author Affiliation: Dept. of Comput. Sci. & Inf. Eng., Inha Univ., Inchon, South Korea
Conference Title: Database and Expert Systems Applications. 17th International Conference, DEXA 2006. Proceedings (Lecture Notes in Computer Science Vol. 4080) p. 430-8
Editor(s): Bressan,S.; Kung,J.; Wagner,R.
Publisher: Springer-Verlag , Berlin, Germany
Publication Date: 2006 **Country of Publication:** Germany xxi+959 pp.
ISBN: 3 540 37871 5 **Material Identity Number:** XX-2006-01433
Conference Title: Database and Expert Systems Applications. 17th International Conference, DEXA 2006. Proceedings
Conference Date: 4-8 Sept. 2006 **Conference Location:** Krakow, Poland

Language: English **Document Type:** Conference Paper (PA)
Treatment: Practical (P); Theoretical (T)
Abstract: Utilizing Global Positioning System (GPS) technology, it is possible to find and **recommend** restaurants for users operating mobile devices. For **recommending** restaurants, **personal digital assistants** or cellular phones only consider the location of restaurants. However, a user's background and environment information is assumed to be directly related to recommendation quality. In this paper, therefore, a **recommender** system using context information and a decision tree model for efficient recommendation is presented. This system considers location context, personal context, environment context, and user preference. Restaurant lists are obtained from location context, personal context, and environment context using the decision tree model. In addition, a **weight** value is used for reflecting user preferences. Finally, the system **recommends** appropriate restaurants to the mobile user. For this experiment, performance was verified using measurements such as k-fold cross-validation and mean absolute error. As a result, the proposed system obtained an improvement in recommendation performance. (15 Refs)
Subfile: C
Descriptors: data mining; decision trees; Global Positioning System; information filtering; information filters; mobile computing
Identifiers: location-based service; restaurant **recommendation system**; Global Positioning System; **personal digital assistant**; cellular phone; context information; decision tree model; k-fold cross-validation; mean absolute error; information filtering; data mining
Class Codes: C7250N (Search engines); C6150N (Distributed systems software); C7250R (Information retrieval techniques); C1160 (Combinatorial mathematics); C6170K (Knowledge engineering techniques)

7/5/13 (Item 2 from file: 2) Links
INSPEC

09243963 **INSPEC Abstract Number:** C2005-02-7250R-091
Title: Personalized image recommendation in the mobile Internet
Author Yoon Ho Cho; Chan Young Kim; Deok Hwan Kim
Author Affiliation: Sch. of e-Bus., Kookmin Univ., Seoul, South Korea
Conference Title: PRICAI 2004: Trends in Artificial Intelligence. 8th Pacific Rim International Conference on Artificial Intelligence. Proceedings (Lecture Notes in Artificial Intelligence Vol.3157) p. 963-4
Editor(s): Zhang, C.; Guesgen, H.W.; Yeap, W.K.
Publisher: Springer-Verlag , Berlin, Germany
Publication Date: 2004 **Country of Publication:** Germany xx+1023 pp.
ISBN: 3 540 22817 9 **Material Identity Number:** XX-2004-02000
Conference Title: PRICAI 2004: Trends in Artificial Intelligence. 8th Pacific Rim International Conference on Artificial Intelligence. Proceedings
Conference Date: 9-13 Aug. 2004 **Conference Location:** Auckland, New Zealand

Language: English **Document Type:** Conference Paper (PA)
Treatment: Practical (P)
Abstract: As mobile Internet technology becomes more increasingly applicable, the mobile contents market, especially character image downloading for **mobile phones**, has recorded remarkable growth. In spite of this rapid growth, however, most of the customers experience inconvenience, lengthy search processes and frustration in searching for the specific character images they want due to inefficient sequential search. This article describes a personalized image **recommender** system designed to reduce customers' search efforts in finding desired character images on the mobile Internet. The system combines two of the most popular information filtering techniques: **collaborative filtering** and content-based image retrieval. (2 Refs)
Subfile: C
Descriptors: content-based retrieval; data handling; groupware; information filtering; Internet; mobile computing; mobile handsets
Identifiers: personalized image recommendation; mobile Internet; mobile contents market; character image downloading; **mobile phones**; lengthy search processes; sequential search; personalized image **recommender** system; information filtering; **collaborative filtering**; content-based image retrieval
Class Codes: C7250R (Information retrieval techniques); C5260B (Computer vision and image processing techniques); C6150N (Distributed systems software); C7210N (Information networks); C6130 (Data handling techniques)

7/5/14 (Item 3 from file: 2) Links
Fulltext available through: custom link USPTO Full Text Retrieval Options
INSPEC

09235851 **INSPEC Abstract Number:** C2005-02-7250N-016
Title: VISCORS: a visual-content recommender for the mobile Web
Author Chan Young Kim; Jae Kyu Lee; Yoon Ho Cho; Deok Hwan Kim
Author Affiliation: Korea Adv. Inst. of Sci. & Technol., Seoul, South Korea
Journal: IEEE Intelligent Systems vol.19, no.6 p. 32-9
Publisher: IEEE ,
Publication Date: Nov.-Dec. 2004 **Country of Publication:** USA
CODEN: IISYF7 **ISSN:** 1541-1672
SICI: 1541-1672(200411/12)19:6L.32:VVCR;1-A
Material Identity Number: G263-2004-006
U.S. Copyright Clearance Center Code: 1541-1672/04/$20.00

Language: English **Document Type:** Journal Paper (JP)
Treatment: Practical (P)
Abstract: Current search methods for mobile-Web content can be frustrating to use. To shorten searches for **cell phone** wallpaper images, VISCORS combines **collaborative filtering** with content-based image retrieval. An increasing selection of content is becoming available in the mobile-Web environment, where users navigate the Web using wireless devices such as **cell phones** and **PDAs**. The fast growth and excellent prospects of the mobile-Web content market have attracted many content providers. (10 Refs)
Subfile: C
Descriptors: content-based retrieval; image retrieval; information filtering; information filters; Internet; mobile computing; user interfaces
Identifiers: VISCORS; visual-content **recommender**; mobile Web environment; mobile-Web content; **cell phone** wallpaper images; **collaborative filtering**; content-based image retrieval; wireless devices
Class Codes: C7250N (Search engines); C7250R (Information retrieval techniques); C6150N (Distributed systems software); C5260B (Computer vision and image processing techniques)

7/5/15 (Item 4 from file: 2) **Links**

Fulltext available through: USPTO Full Text Retrieval Options

INSPEC

07876229 **INSPEC Abstract Number:** C2001-05-7180-004

Title: Personalization of supermarket product recommendations

Author Lawrence, R.D.; Almasi, G.S.; Kotlyar, V.; Viveros, M.S.; Duri, S.S.
Author Affiliation: IBM Thomas J. Watson Res. Center, Yorktown Heights, NY, USA

Journal: Data Mining and Knowledge Discovery vol.5, no.1-2 p. 11-32
Publisher: Kluwer Academic Publishers ,
Publication Date: 2001 **Country of Publication:** Netherlands
CODEN: DMKDFD **ISSN:** 1384-5810
SICI: 1384-5810(2001)5:1/2L.11:PSPR;1-Y
Material Identity Number: G116-2001-001
U.S. Copyright Clearance Center Code: 1384-5810/2000/$19.50

Language: English **Document Type:** Journal Paper (JP)

Treatment: Practical (P)

Abstract: Describes a personalized **recommender** system that has been designed to suggest new products to supermarket shoppers. The **recommender** functions in a pervasive computing environment, namely a remote shopping system in which supermarket customers use **personal digital assistants (PDAs)** to compose and transmit their orders to the store, which assembles them for subsequent pickup. The **recommender** is meant to provide an alternative source of new ideas for customers who now visit the store less frequently. Recommendations are generated by matching products to customers based on the expected appeal of the product and the previous spending of the customer. Association mining in the product domain is used to determine relationships among product classes for use in characterizing the appeal of individual products. Clustering in the customer domain is used to identify groups of shoppers with similar spending histories. Cluster-specific lists of popular products are then used as input to the matching process. The **recommender** is currently being used in a pilot program with several hundred customers. Analysis of the results to date have shown a 1.8% boost in program revenue as a result of purchases made directly from the list of **recommended** products. A substantial fraction of the accepted recommendations are from product classes new to the customer, indicating a degree of willingness to expand beyond present purchase patterns in response to reasonable suggestions. (21 Refs)

Subfile: C

Descriptors: data mining; home shopping; microcomputer applications; **notebook** computers; pattern clustering; user modelling

Identifiers: supermarket product recommendations; personalized **recommender** system ; new product suggestions; supermarket shoppers; pervasive computing environment; remote shopping system; **personal digital assistants**; order composition; order transmission; product-customer matching; product appeal; previous customer spending; association mining; product class relationships; customer domain clustering; shopper spending histories; cluster-specific product lists; pilot program; revenue; purchase patterns; **collaborative filtering**

Class Codes: C7180 (Retailing and distribution computing); C7830 (Home computing); C6170K (Knowledge engineering techniques); C6180 (User interfaces)

7/5/16 (Item 5 from file: 2) **Links**
INSPEC

07747873 **INSPEC Abstract Number:** C2000-12-7180-007
Title: A PDA-based personalized recommender agent
Author Almasi, G.S.; Lee, A.J.
Author Affiliation: IBM Thomas J. Watson Res. Center, Yorktown Heights, NY, USA
Conference Title: Proceedings of the Fifth International Conference on the Practical Application of Intelligent Agent and Multi Agent Technology p. 299-309
Publisher: Practical Application Company , Blackpool, UK
Publication Date: 2000 **Country of Publication:** UK 400 pp.
ISBN: 1 902426 07 X **Material Identity Number:** XX-2000-00749
Conference Title: Proceedings of PAAM 2000
Conference Date: 10-12 April 2000 **Conference Location:** Manchester, UK

Language: English **Document Type:** Conference Paper (PA)
Treatment: Practical (P)
Abstract: As an example of distributed personalization in a pervasive computing environment, this paper describes a **PDA (personal digital assistant)** based personal "wine guru" agent that works hand-in-hand with an on-board supermarket shopping program, and also with a server-based data-mining program that provides personalised wine lists. The guru ascertains which list of the store's wines from the server best matches the user's tastes in wine, keeps an eye on the user's on-board wine-cellar list, and can read the shopping list that the user is preparing and then suggest wines to go with some of the items after asking how they will be prepared. The user may elect to add some of the suggested wines to the shopping list, or choose from the wine cellar. For portability, a subset of Java with a virtual machine small enough to fit on to a **PalmPilot** was used. It provided all the needed functionality and an acceptable response time. (12 Refs)
Subfile: C
Descriptors: data mining; microcomputer applications; **notebook** computers; personal computing; point of sale systems; software agents; software portability; user modelling; virtual machines
Identifiers: **PDA**-based personalized **recommender** agent; distributed personalization; pervasive computing environment; **personal digital assistant**; wine guru; personal agent; on-board supermarket shopping program; server-based data-mining program; personalised wine lists; user tastes; on-board wine-cellar list; shopping list; food preparation; software portability; Java subset; virtual machine; **PalmPilot**; functionality; response time; **collaborative filtering**; content filtering; intelligent agents; intelligent assistant; Waba; clustering
Class Codes: C7180 (Retailing and distribution computing); C6170 (Expert systems and other AI software and techniques); C6180 (User interfaces)

7/5/18 (Item 1 from file: 34) Links
Fulltext available through: Institute of Electrical and Electronics Engineers USPTO Full Text Retrieval Options
SciSearch(R) Cited Ref Sci

13391261 **Genuine Article#:** 873MP **Number of References:** 10
Visors: A visual-content recommender for the mobile Web

Author: Kim CY (REPRINT) ; Lee JK; Cho YH; Kim DH
Corporate Source: Korea Adv Inst Sci & Technol,Grad Sch Management,207-43 Cheongryangri/Seoul 130012//South Korea/ (REPRINT); Korea Adv Inst Sci & Technol,Grad Sch Management,Seoul 130012//South Korea/; Dongyang Tech Coll,Dept Internet Business,Seoul 152714//South Korea/; Kookmin Univ,Sch EBusiness,Seoul 136702//South Korea/; Dongyang Tech Coll,Dept Mobile Internet,Seoul 152714//South Korea/ (cykim@dongyang.ac.kr; jklee@kgsm.kaist.ac.kr; www4u@kookmin.ac.kr; dhkim@dongyang.ac.kr)
Journal: IEEE INTELLIGENT SYSTEMS , 2004 , V 19 , N6 (NOV-DEC) , P 32-39
ISSN: 1094-7167 **Publication date:** 20041100
Publisher: IEEE COMPUTER SOC , 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1314 USA
Language: English **Document Type:** ARTICLE
Geographic Location: South Korea
Journal Subject Category: COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE; ENGINEERING, ELECTRICAL & ELECTRONIC
Abstract: Current search methods for mobile Web content can be frustrating to use. To shorten searches for **cellphone** wallpaper images, ViscoRs combines **collaborative filtering** with content based image retrieval.
Cited References:
*KOR NETW INF CTR, 2002, KOR INT WHIT PAP 200
BALABANOVIC M, 1997, V40, P66, COMMUN ACM
CHO YH, 2002, V23, P329, EXPERT SYST APPL
KIM DH, 2003, P599, P ACM SIGMOD INT C
MELVILLE P, 2002, P187, P 18 NAT C ART INT
PORKAEW K, 1999, P233, P 7 ACM INT MULT C 1
SARWAR B, 2000, P158, P 2 ACM C EL COMM
SHARDANAND U, 1995, P210, P C HUM FACT COMP SY
WU L, 2000, P297, P 26 INT C VER LARG
ZHOU XS, 2003, V8, P536, ACM MULTIMEDIA SYSTE

7/5/19 (Item 1 from file: 95) **Links**

01830726 20040206025

Cellular phone ringing tone recommendation system based on collaborative filtering method

Kostov, V; Naito, E; Ozawa, J
Adv. Technol. Res. Lab., Matsushita Electr. Ind. Comp., Kyoto, J

2003 IEEE Internat. Symp. on Computational Intelligence in Robotics and Automation, Computational Intelligence in Robotics and Automation for the New Millennium, Proc., Vol. 1, Kobe, JP, 16-20 July 2003 , 2003
Document type: Conference paper **Language:** English
Record type: Abstract
ISBN: 0-7803-7866-0

Abstract:
We have developed a prototype of cellular phone ringing tome **recommendation system** using memory-based **collaborative filtering** and we have carried out examinations to **evaluate** its performance. The ringing tone content was stored on a server from where the users were able to download the desired items according to their preferences. An extensive log data accumulated at the download service site for a fixed period of time was used. The log data contained only information for the users' downloaded ringing tomes without **evaluation** data. The user set and the tone downloadable content set were not fixed and our goal was to investigate how **collaborative filtering** could be successfully applied to a system with such continuously changing conditions. The Jaccard's similarity coefficient was used to calculate the similarity between the users. The learning period, the recommendation period and the number of the similar users were used as condition parameters. The system quality **evaluation** showed that the recall increases with the increase of the learning period but decreases with the increase of the recommendation period. Optimal values for the number of the most similar users as well as for the learning and the recommendation periods were experimentally obtained. It was shown that the **collaborative filtering** method could be successfully applied to a cellular phone ringing tone **recommendation system**.

Descriptors: CELLULAR RADIO; INFORMATION RETRIEVAL SYSTEMS; LEARNING SYSTEMS; **CELL PHONES**; CLIENT SERVER SYSTEMS
Identifiers: ZELLULARTELEFON; Zellularfunk; Informationswiedergewinnung

13/5/1 (Item 1 from file: 2) **Links**
INSPEC

08176285 **INSPEC Abstract Number:** C2002-03-6170K-048
Title: Collaborative filtering for a distributed smart IC card system
Author Murakami, E.; Terano, T.
Author Affiliation: Yamatake Corp., Tokyo, Japan
Conference Title: Intelligent Agents: Specification, Modeling, and Applications. 4th Pacific Rim International Workshop on Multi-Agents, PRIMA 2001. Proceedings (Lecture Notes in Artificial Intelligence Vol.2132) p. 183-97
Editor(s): Yuan, S-T; Yokoo, M.
Publisher: Springer-Verlag , Berlin, Germany
Publication Date: 2001 **Country of Publication:** Germany vii+236 pp.
ISBN: 3 540 42434 2 **Material Identity Number:** XX-2001-02370
Conference Title: Intelligent Agents: Specification, Modeling, and Applications. 4th Pacific Rim International Workshop on Multi-Agents
Conference Date: 28-29 July 2001 **Conference Location:** Taipei, Taiwan

Language: English **Document Type:** Conference Paper (PA)
Treatment: Practical (P)
Abstract: Collaborative filtering, often used in E-commerce applications, is a method to cluster similar users based on their profiles, characteristics or attitudes on specific subjects. This paper proposes a novel method to implement dynamic **collaborative filtering** by Genetics-based machine learning, in which we employ Learning Classifier Systems extended to multiple environments. The proposed method is used in a yet another **mobile** agent system: a distributed smart IC card system. The characteristics of the proposed method are summarized as follows: (1) It is effective in **distributed computer** environments with PCs even for small number of users. (2) It learns users' profiles from the individual behaviors of them then generates the recommendation and advices for each user. (3) The results are automatically accumulated in a local system on a PC, then they are distributed via smart IC cards while the users are interacting with the system. The method has been implemented and validated in Group Trip Advisor prototype: a PC-based **distributed recommendor** system for travel information. (12 Refs)
Subfile: C
Descriptors: learning (artificial intelligence); pattern clustering; smart cards; software agents; travel industry
Identifiers: dynamic **collaborative filtering**; Learning Classifier Systems; **mobile** agent; distributed smart IC card; travel information; Group Trip Advisor
Class Codes: C6170K (Knowledge engineering techniques); C6170 (Expert systems and other AI software and techniques); C7185 (Administration of other service industries)

16/5/1 (Item 1 from file: 35) **Links**
Dissertation Abs Online

01741769 ORDER NO: AADAA-I9968225
User model induction for intelligent information access

Author: Billsus, Daniel-Alexander
Degree: Ph.D.
Year: 2000
Corporate Source/Institution: University of California, Irvine (0030)
Chair: Michael J. Pazzani
Source: Volume 6104B of Dissertations Abstracts International.
PAGE 2029 . 265 PAGES
Descriptors: COMPUTER SCIENCE ; ARTIFICIAL INTELLIGENCE
Descriptor Codes: 0984; 0800

The explosive growth of information available on the Internet has created a clear need for novel methods that help users locate relevant information quickly and with minimal effort. The central argument of this dissertation is that learning about users' multiple and potentially changing interests calls for algorithms specifically designed for this purpose. Guided by this principle, I introduce the <italic>Adaptive Information Server</italic> (<italic> AIS</italic>), a client-server framework for domain-independent adaptive information access. I describe two applications that use <italic>AIS</italic> to learn about users' interests in daily news stories: one system operates on the World Wide Web, the other is geared towards **wireless** information access. The description and evaluation of the underlying recommendation algorithms form the core of the dissertation.

First, I describe a content-based learning algorithm designed to learn about users' multiple and frequently changing interests. The key to the algorithm's performance lies in its multi-strategy design: it learns separate models of users' short-term and long-term interests. An empirical **evaluation** shows that the **combination** of both models performs better than each individual model alone. In addition, the algorithm maintains a model of information the user is likely to know, so that the presentation of redundant content can be avoided. Second, I show how the described content-based algorithm can be extended with a **collaborative filtering** component. In particular, I cast **collaborative filtering** as a learning task, and present a novel algorithm that uses the Singular Value Decomposition to derive a low-dimensional data representation that forms the basis of an efficient and accurate approach to **collaborative filtering**. Empirical results demonstrate that the resulting approach outperforms previously proposed algorithms, and that combining content-based and collaborative techniques leads to overall performance improvements.

The dissertation concludes with a description of two empirical studies that evaluate the utility of adaptive information access from a user perspective. These studies show that the adaptive presentation of personalized content simplifies access to relevant information, and that the observed performance can be achieved without requiring any extra work from the user.

16/5/3 (Item 1 from file: 144) **Links**
Fulltext available through: USPTO Full Text Retrieval Options ProQuest
Pascal

17467927 PASCAL No.: 06-0051378

Making better recommendations with online profiling agents
Innovative Artificial Intelligence Applications

CHIN HOCK OH Danny; CHEW LIM TAN
HILL Randall W, ed; JACOBSTEIN Neil, ed
National University of Singapore, Singapore; Department of Computer Science, School of Computing, National University of Singapore, Singapore
Annual Conference on Innovative Applications of Artificial Intelligence, 16 (San Jose, California USA) 2004-07-27
Journal: The AI magazine, 2005
, 26 (3) 29-39

ISSN: 0738-4602 Availability: INIST-22053; 354000132699400020
No. of Refs.: 11 ref.
Document Type: P (Serial); C (Conference Proceedings) ; A (Analytic)
Country of Publication: United States
Note: 4 notes
Language: English
In recent years, we have witnessed the success of autonomous agents applying machine-learning techniques across a wide range of applications. However, agents applying the same machine-learning techniques in online applications have not been so successful. Even agent-based hybrid **recommender** systems that **combine** information filtering techniques with **collaborative filtering** techniques have been applied with considerable success only to simple consumer goods such as movies, books, clothing, and food. Yet complex, adaptive autonomous agent systems that can handle complex goods such as real estate, vacation plans, insurance, mutual funds, and mortgages have emerged. To a large extent, the reinforcement learning methods developed to aid agents in learning have been more successfully deployed in offline applications. The inherent limitations in these methods have rendered them somewhat ineffective in online applications. In this article, we postulate that a small amount of prior knowledge and human-provided input can dramatically speed up online learning. We demonstrate that our agent HumanE-with its prior knowledge or "experiences" about the real estate domain-can effectively assist users in identifying requirements, especially unstated ones, quickly and unobtrusively.

English Descriptors: Autonomous system; Multiagent system; Artificial intelligence; **Remote** teaching; Information extraction; Complex system; Intelligent agent; Adaptive system; Reinforcement learning; Recommendation; Consumer good; Insurance; User requirement; User need; Filtering
French Descriptors: Systeme autonome; Systeme multiagent; Intelligence

artificielle; Teleenseignement; Extraction information; Systeme complexe; Agent intelligent; Systeme adaptatif; Apprentissage renforce; Recommandation; Bien consommation; Assurance; Exigence usager; Besoin de l'utilisateur; Filtrage; .

Classification Codes: 001D02C02

Expert Systems with Applications
Volume 27, Issue 2, August 2004, Pages 203-210

doi:10.1016/j.eswa.2004.01.003 Cite or Link Using DOI

A scalable P2P recommender system based on distributed collaborative filtering

Peng Han, Bo Xie, Fan Yang and Ruimin Shen
Department of Computer Science and Engineering, Shanghai Jiaotong University, 6th Floor Haoran High-tech Building, Shanghai 200030, China

Available online 10 February 2004.

Abstract

Collaborative Filtering (CF) technique has been proved to be one of the most successful techniques in recommender systems in recent years. However, most existing CF based recommender systems worked in a centralized way and suffered from its shortage in scalability as their calculation complexity increased quickly both in time and space when the record in user database increases. In this article, we first propose a distributed CF algorithm called PipeCF together with two novel approaches: significance refinement and unanimous amplification, to further improve the scalability and prediction accuracy. We then show how to implement this algorithm on a Peer-to-Peer (P2P) structure through distributed hash table method, which is the most popular and efficient P2P routing algorithm, to construct a scalable distributed recommender system. The experimental data show that the distributed CF-based recommender system has much better scalability than traditional centralized ones with comparable prediction efficiency and accuracy.

Author Keywords: Recommender system; Collaborative filtering; Peer-to-Peer; Significance refinement; Unanimous amplification

Article Outline

1. Introduction

Recommender system is a system that helps users to find their wanted items by making recommendations based on either the content of the recommended items (Content-based Filtering), or ratings of similar users on the recommended items (Collaborative Filtering, CF). Since Goldberg, Nichols, Oki, and Terry (1992) published the first account of using CF for information filtering; CF has proved to be one of the most successful techniques in recommendation systems by its advantage of that no explicit description of items is needed. The key idea of CF is that users will prefer those items that people with similar interests prefer, or even that dissimilar people do not prefer, so most CF algorithms can be separated into three steps as addressed by Herlocker, Konstan, Borchers, and Riedl (1999): (1) Similarity Weight: weight all users with respect to similarity with the active user, which refer to the user whose preferences are to be predicted; (2) Selecting Neighborhoods: select those users used to make prediction; (3) Rating Normalization and Prediction Making: normalize and calculate the weighted sum of selected users' ratings, then make prediction based on that. According to different techniques used in the first part mentioned above, CF algorithms can be divided into two classes: memory-based algorithms and model-based algorithms. Breese et al. performed an empirical analysis on both of two kinds of CF algorithms in Breese, Heckerman, and Kadie (1998) while Herlock et al. presented an algorithmic framework for performing CF in Herlocker et al. (1999).

GroupLens (Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994) was the first CF algorithm to automate prediction and used a memory-based algorithm. Like most memory-based algorithms, GroupLens need to

compute across the whole user database to calculate the similarities between active user and other users to make prediction. Ringo (Shardanand & Maes, 1995) only used those neighbors whose correlation were greater than a given threshold to make prediction. This approach not only reduced the calculation complexity but also proved to improve the performance. By choosing top-N users with the highest correlations the same improvement can also been obtained. However, all the other users' similarities still have to be calculated and its complexity increased quickly both in time and space as the record in the database increases.

Basically, there are two ways to reduce this calculation complexity. The first one is used a model-based algorithm which first constructs some certain mathematical models, such as Bayesian Network, Bayesian Classifiers et al., to describe the users and/or their ratings, then learns these models from the database and use them to make prediction. However, these approaches also need complex calculation when compiling models and also require a central database to keep all the user data which is not easy to achieve sometime not only for techniques reasons but also for privacy reasons.

The second way is to implement CF in a decentralized way. In fact, as Peer-to-Peer (P2P) gains more and more popularity, some researchers have already begun to consider it as an alternative architecture to reduce the calculation complexity (Tveit, 2001; Olsson, 2003 and Canny, 2002) of centralized CF algorithms. The main difference between centralized CF-based recommender system and distributed ones is that the originally centralized user database are maintained in a decentralized way which means each peer will only keep a fraction of user database and when making prediction for a particular user, needed record should first be retrieved to the user's own database from other peers and calculated locally. In order to do this, the following two problems have to be addressed: (1) how to store the user database distributed efficient so that needed information can be found efficiently; (2) how to identify those records needed to make prediction for a particular user and fetch them efficiently as retrieving all other users' votes back is not only unreasonable but also unnecessary.

The main contributions of this article are:

(1) We propose PipeCF: a distributed CF algorithm which can be implemented on a P2P overlay network;
(2) We propose two novel approaches: significance refinement (SR) and unanimous amplification (UA), to improve the performance of our distributed CF algorithm;
(1) We give the framework of implementing our distributed CF algorithm on P2P overlay network through distributed hash table (DHT) based technique to obtain efficient user database management and retrieval to construct decentralized CF recommender system.

The rest of this article is organized as follows. In Section 2, several related works are presented and discussed. In Section 3, we introduce the architecture and key features of our DHT-based CF system. Two techniques: SR and UA are also proposed in this section to improve the scalability and prediction accuracy of DHT-based CF algorithm in this section. In Section 4 the experimental results of our system are presented and analyzed. Finally we make a brief concluding remark and give the future work in Section 5.

2. Related work

2.1. Memory-based CF algorithm

Generally, the task of CF is to predict the votes of active users from the user database which consists of a set of votes $v_{i,j}$ corresponding to the vote of user *i* on item *j*. Memory-based CF algorithm calculates this prediction of as a weighted average of other users votes on that item through the following formula:

$$P_{a,j} = \bar{v}_a + \kappa \sum_{i=1}^{n} \varpi(a,j)(v_{i,j} - \bar{v}_i) \tag{1}$$

where $P_{a,j}$ denotes the prediction of the vote for active user a on item j and n is the number of users in user database. $\bar{v}_i$ is the mean vote for user i as:

$$\bar{v}_i = \frac{1}{|I_i|} \sum_{j \in I_i} v_{i,j} \tag{2}$$

where I_i is the set of items on which user i has voted. The weights $\varpi(a,j)$ reflect the similarity between active user and users in the user database. κ is a normalizing factor to make the absolute values of the weights sum to unity.

Most memory-based algorithms use Eq. (1) to make prediction and only distinguish between the ways they calculate the weights:

2.1.1. Pearson correlation coefficient

Pearson correlation coefficient was first introduced into collaborative filtering as a weighting method in the GroupLens project The correlation between user a and i is:

$$\varpi(a,i) = \frac{\sum_j (v_{a,j} - \bar{v}_a)(v_{i,j} - \bar{v}_i)}{\sqrt{\sum_j (v_{a,j} - \bar{v}_a)^2 \sum_j (v_{i,j} - \bar{v}_i)^2}} \tag{3}$$

where the summations is calculated over those items for which both users a and i have voted.

2.1.2. Vector similarity

The vector similarity was first used to measure the similarity between two documents. Each document was viewed as a vector of word frequency and their similarity was computed as the cosine of the angle between these two vectors. In Collaborative Filtering, we treat each user record as a document and their votes as frequency of items. So the weights can now be calculated as:

$$\varpi(a,i) = \sum_j \frac{v_{a,j}}{\sqrt{\sum_{k \in I_a} v_{a,k}^2}} \frac{v_{i,j}}{\sqrt{\sum_{k \in I_i} v_{i,k}^2}} \tag{4}$$

2.2. P2P system and DHT routing algorithm

The term 'Peer-to-Peer' refers to a class of systems and applications that employ distributed resources to perform a critical function in a decentralized manner. With the pervasive deployment of computers, P2P is increasingly receiving attention in research and more and more P2P systems have been deployed on the Internet. Some of the benefits of a P2P approach include: improving scalability by avoiding dependency on centralized points; eliminating the need for costly infrastructure by enabling direct communication among clients; and enabling resource aggregation. Among all these applications, three main classes of peer-to-peer applications have emerged: parallelizable, content and file management, and collaborative.

As the main purpose of P2P systems are to share resources among a group of computers called peers in a distributed way, efficient and robust routing algorithms for locating wanted resource is critical to the performance of P2P systems. Among these algorithms, distributed hash table (DHT) algorithm is one of the most popular and

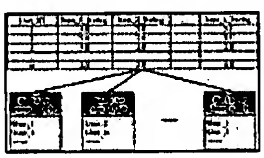
effective and supported by many P2P systems such as CAN (Ratnasamy, Francis, Handley, Karp, & Shenker, 2001), Chord (Stocal et al., 2001), Pastry (Rowstron & Druschel, 2001), and Tapestry (Zhao et al., 2001).

A DHT overlay network is composed of several DHT nodes and each node keeps a set of resources (e.g. files, rating of items). Each resource is associated with a key (produced, for instance, by hashing the file name) and each node in the system is responsible for storing a certain range of keys. Peers in the DHT overlay network locate their wanted resource by issue a *lookup(key)* request which returns the identity (e.g. the IP address) of the node that stores the resource with the certain key. The primary goals of DHT are to provide an efficient, scalable, and robust routing algorithm which aims at reducing the number of P2P hops, which are involved when we locate a certain resource, and to reduce the amount of routing state that should be preserved at each peer. In Chord (Stocal et al., 2001), each peer keeps track information of *logN* other peers (*N* is the total number of peers in the community). When a peer joins and leaves the overlay network, this highly optimized version of DHT algorithm will only require notifying *logN* peers about that change.

3. Our distributed CF algorithm

3.1. Basic PipeCF algorithm

The first step to implement CF algorithm in a distributed way is to divide the original centralized user database into fractions which can be stored in Peers distributed For concision, we use the term *bucket* to denote one fraction of the whole user database in the following of this article. Each bucket should also be assigned an identifier through which they can be located later when needed. The way we do the division is to make each bucket hold a group of users' record who has at least rated one item with the same vote. By that we construct one bucket for every different <ITEM_ID, VOTE> tuple and use that tuple as the identifier for the bucket. Fig. 1 shows our division strategy:



Display Full Size version of this image (4K)

Fig. 1. User database division strategy.

In order to reduce the calculation complexity and achieve scalability, we wish to find a strategy which can choose neighbors from the most suitable buckets when making prediction. The PipeCF algorithm chooses neighbors based on the heuristic that people with similar interests at least rate one item with similar votes. As we can see in Fig. 2, this strategy have very high hitting ratio. So when making prediction, the PipeCF only uses those users' records who are in at least one same bucket with the active users. Through which we reduce about 50% calculation than traditional CF algorithm and obtain comparable prediction as shown in Figure 8.

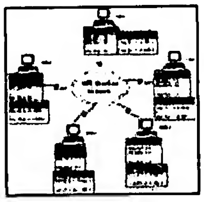
 **Display Full Size version of this image** (8K)

Fig. 2. Architecture of DHT-based CF recommender system.

3.2. Improved PipeCF algorithm

3.2.1. Significance refinement

In the basic PipeCF algorithm, we use all users which are in the same bucket with the active user and find that the algorithm has an O(*N*) fetched user number (*N* is the total user number) as Fig. 8 shows. In fact, as Breese et al. (1998) presented by the term inverse user frequency, universally liked items are not as useful as less common items in capturing similarity. So we introduce a new concept SR, which reduces the returned user number of the basic PipeCF algorithm by limiting the number of returned users for each bucket. We term the algorithm improved by SR as Return K which means 'for every item, the PipeCF algorithm returns no more than K users for each bucket'. The experimental result in Section 5.3.3 shows that this method reduces the returned user number dramatically and also improves the prediction accuracy.

3.2.2. Unanimous amplification

Enlightened by the method of case amplification (Breese et al., 1998) which emphasizes the contribution of the most similar users to the prediction by amplifying the weights close to 1, we argue that we should give special award to the users who rated some items with the same vote by amplify their weights, which we term UA We transform the estimated weights as follows:

$$w'_{a,i} = \left\{ \begin{array}{ll} w_{a,i} & N_{a,i} = 0 \\ w_{a,i}\alpha & 0 < N_{a,i} \le \gamma \\ w_{a,i}\beta & N_{a,i} > \gamma \end{array} \right\} \tag{5}$$

where $N_{a,i}$ denotes the number of items which user *a* and user *i* have the same votes. A typical value for α for our experiments is 2.0, β is 4.0, and γ is 4. Experimental result in Section 4.3.4 shows that UA approach improves the prediction accuracy of the PipeCF algorithm.

4. DHT-BASED CF recommender system

4.1. Architecture of DHT-based CF recommender system

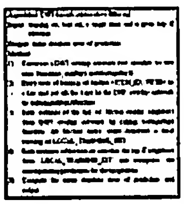
The main advantage of our DHT-based CF recommender system vs. traditional centralized CF recommender system is that both the maintenance of user database and the complex computation task of making prediction are done in a decentralized way so as to obtain better scalability. Here, we treat each bucket as resource and the unique key for the resource is generated by the particular <ITEM_ID, VOTE> tuple associated with the bucket. Each peer in the DHT overlay network will keep one or several buckets locally. Fig. 2 gives the architecture of our DHT-based CF recommender system.

So the implementation of PipeCF in DHT overlay network is straightforward except that the bucket is stored

distributed so that when a user wants to look up other similar users which have the same particular <ITEM_ID, VOTE> tuple, it need fetch them from DHT overlay network. Still we need special scheme to manage the distributed storage of buckets. DHT has provided two function *lookup(key)* and *put(key)* which we will describe later to do these jobs and its efficiency has been guaranteed by the algorithm itself. So with the DHT overlay network, all the users in the CF system are connected together and can find their wanted similar neighbors efficiently through a DHT routing algorithm.

4.2. Implementation of PipeCF on DHT

On the basis of the decentralized storage of user votes, we introduce our implementation of PipeCF algorithm on the DHT overlay network, called DHT-based CF algorithm, as shown in Fig. 3.



Display Full Size version of this image (17K)

Fig. 3. DHT-based CF algorithm.

There are two key pieces to the DHT-based CF system algorithm: the lookup mechanism used to locate similar users and fetch their actual rating. The decentralized storage (and hence decentralized retrieval) in decentralized CF system makes the CF calculation inherently scalable (every user do recommendation locally instead of depending on a centralized server); the hard part is finding the similar peers from which to retrieve the actual rating.

We devise a scalable solution to the problem of locating similar users in decentralized CF system, i.e. give a user vote vector; we can find the IP address of the node(s) which is similar to the user. Our DHT-based solution can reach following goal:

• *Scalability*: it must be designed to scale to several million nodes.
• *Efficiency*: similar users should be located reasonably quick and with low overhead in terms of the message traffic generated.
• *Dynamicity*: the system should be robust to frequent node arrivals and departures in order to cope with highly transient user populations' characteristic to decentralized environments.
• *Balanced load*: in keeping with the decentralized nature, the total resource load (traffic, storage, etc) should be roughly balanced across all the nodes in the system.

So we only select similar users in the subset in which users have same <ITEM_ID, VOTE> tuple. The key idea of our algorithm is hashing every user for every rated item. Our DHT-based CF algorithm includes two main DHT functions: *put(key)* and *lookup(key)*, and Fig. 4 and Fig. 5 show them separately.

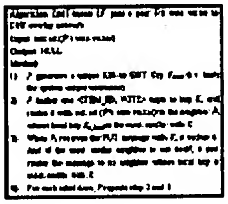

Display Full Size version of this image (15K)

Fig. 4. DHT-based CF put function.

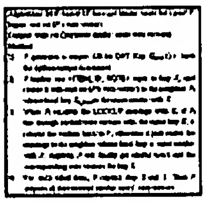

Display Full Size version of this image (19K)

Fig. 5. DHT-based CF lookup function.

DHT-based CF Put algorithm is used to construct DHT overlay network and fill data in it. DHT-based CF Lookup algorithm is used to lookup and fetch similar uses with same <ITEM_ID, VOTE> tuple in order to construct a local training set to make recommendation. The main purpose of steps 2 and 3 in Fig. 3 is to make every peer in the DHT overlay network keep several buckets which contain a group of users with same <ITEM_ID, VOTE> tuple, from which the Lookup algorithm can fetch similar users later in its steps 2 and 3.

5. Experimental evaluation

In this section, we describe the dataset, metrics and methodology for the comparison between traditional and DHT-based CF algorithm, and present the results of our experiments.

5.1. Data set

We use Eachmovie collaborative filtering data set (1997) to evaluate the performance of improved algorithm The EachMovie data set is provided by the Compaq System Research Center, which ran the EachMovie recommendation service for 18 months to experiment with a collaborative filtering algorithm. The information they gathered during that period consists of 72,916 users, 1628 movies, and 2,811,983 numeric ratings ranging from 0 to 5. To speed up our experiments, we only use a subset of the EachMovie data set.

5.2. Metrics and methodology

The metrics for evaluating the accuracy of a prediction algorithm can be divided into two main categories: statistical accuracy metrics and decision-support metrics. Statistical accuracy metrics evaluate the accuracy of a predictor by comparing predicted values with user-provided values. Decision-support accuracy measures how well predictions help user select high-quality items. We use Mean Absolute Error (MAE), a statistical accuracy metrics, to report prediction experiments for it is most commonly used and easy to understand:

$$\mathrm{MAE} = \frac{\sum_{a\in T} |v_{a,j} - p_{a,j}|}{|T|} \qquad (6)$$

where v_{aj} is the rating given to item j by user a, is the predicted value of user a on item j, T is the test set, $|T|$ is the size of the test set.

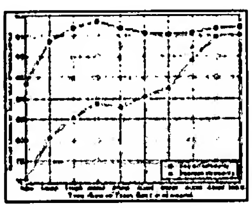
We select 2000 users and choose one user as active user per time and the remainder users as his candidate neighbors, because every user only make self's recommendation locally. We use the mean prediction accuracy of all the 2000 users as the system's prediction accuracy. For every user's recommendation calculation, our tests are performed using 80% of the user's ratings for training, with the remainder for testing.

5.3. Experimental result

We design several experiments for evaluating our algorithm and analyze the effect of various factors (e.g. SR and UA, etc.) by comparison. All our experiments are run on a Windows 2000-based PC with Intel Pentium 4 processor having a speed of 1.8 GHz and 512 MB of RAM.

5.3.1. The efficiency of neighbor choosing

We used a data set of 5000 users and show among the users chosen by PipeCF algorithm, how many are in the top-100 users in Fig. 6. We can see from the data that when the user number rises above 1000, more than 80 users who have the most similarities with the active users are chosen by PipeCF algorithm.

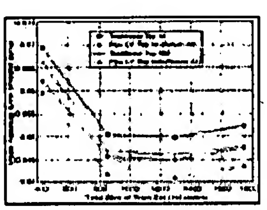


Display Full Size version of this image (5K)

Fig. 6. How many users of PipeCF in traditional CF's top 100.

5.3.2. Performance comparison

We compare the prediction accuracy of traditional CF algorithm and PipeCF algorithm while we apply both top-all and top-100 user selection on them. The results are shown as Fig. 7. We can see that the DHT-based algorithm has better prediction accuracy than the traditional CF algorithm.



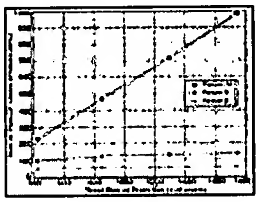
Display Full Size version of this image (5K)

Fig. 7. PipeCF vs. traditional CF.

5.3.3. The effect of significance refinement

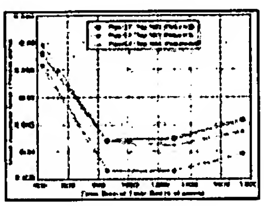
We limit the number of returned user for each bucket by 2 and 5 and do the experiment in Section 5.3.2 again. The user for each bucket is chosen randomly. The result of the number of user chosen and the prediction accuracy is shown in Fig. 8 and Fig. 9, respectively. The result shows:

(1) 'Return All' has an *O(N)* returned user number and its prediction accuracy is also not satisfying;
(2) 'Return 2' has the least returned user number but the worst prediction accuracy;
(1) 'Return 5' has the best prediction accuracy and the scalability is still reasonably well (the returned user number is still limited to a constant as the total user number increases).



Display Full Size version of this image (5K)

Fig. 8. The effect on scalability of SR on PipeCF.

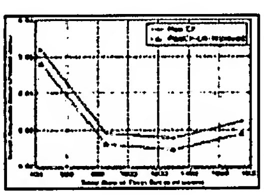


Display Full Size version of this image (5K)

Fig. 9. The effect on prediction accuracy of SR on PipeCF algorithm.

5.3.4. The effect of unanimous amplification

We adjust the weights for each user by using Eq. (5) while setting value for α as 2.0, β as 4.0, γ as 4 and do the experiment in Section 4.3.2 again. We use the top-100 and 'Return All' selection method. The result shows that the UA approach improves the prediction accuracy of both the traditional and the PipeCF algorithm. From Fig. 10 we can see that when UA approach is applied, the two kinds of algorithms have almost the same performance.



Display Full Size version of this image (4K)

Fig. 10. The effect on prediction accuracy of unanimous amplification.

6. Conclusion and future work

In this article, we propose a novel distributed hash table (DHT) based technique to implement efficient user

database management and retrieval in decentralized CF system. Then we propose a heuristic algorithm to fetch similar users from DHT overlay network and do recommendation locally. Finally, we propose two novel approaches: SR and UA to improve the performance of our DHT-based CF algorithm. The experimental data show that our DHT-based CF system has better prediction accuracy, efficiency and scalability than traditional CF systems.

Our future work includes investigation on a more efficient decentralized user database management and K-Nearest Neighbor (KNN) methods which can dynamically self-organize users with semantic similar interests combining content-based filtering techniques. We would also like to investigate on the influence of parameters choosing in UA.

Acknowledgements

The work described in this article is supported partially by National Natural Science Foundation of China under Grant No. 60372078.

References

Breese, J., Heckerman, D. and Kadie, C., 1998. Empirical analysis of predictive algorithms for collaborative filtering. *In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pp. 43–52.

Canny, J., 2002. *Collaborative filtering with privacyIn: Proceedings of the IEEE Symposium on Research in Security and Privacy, Oakland, CA*, IEEE Computer Society, Technical Committee on Security and Privacy, IEEE Computer Society Press pp. 45–57 .

Eachmovie collaborative filtering data set, (1997). http://research.compaq.com/SRC/eachmovie.

Goldberg, D., Nichols, D., Oki, B.M. and Terry, D., 1992. Using collaborative filtering to weave an information tapestry. *Communications of the ACM* **35** 12, pp. 61–70. **Full Text** via CrossRef

Herlocker, J.L., Konstan, J.A., Borchers, A. and Riedl, J., 1999. An algorithmic framework for performing collaborative filtering. *In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 230–237. **Full Text** via CrossRef

Olsson, T (2003). *Bootstrapping and Decentralizing Recommender Systems*. Licentiate Thesis 2003-006. Department of Information Technology, Uppsala University and SICS.

Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S., 2001. A scalable content-addressable network. *In: SIGCOMM.*

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J., 1994. *GroupLens: an open architecture for collaborative filtering of netnewsIn: Proceedings of the 1994 ACM conference on Computer supported cooperative work, October 22–26, 1994, Chapel Hill, North Carolina, United States* pp. 175–186 .

Rowstron, A. and Druschel, P., 2001. *Pastry: Scalable, distributed object location and routing for large scale peer-to-peer systemsIn: IFIP/ACM Middleware*, Hedelberg, Germany.

Shardanand, U. and Maes, P., 1995. *Social information filtering: algorithms for automating 'word of mouth'In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, May 07–11, 1995, Denver, Colorado, United States* pp. 210–217 .

Stocal, I. *et al.*, 2001. *Chord: a scalable peer-to-peer lookup service for Internet applicationsIn: ACM SIGCOMM, San Diego, CA, USA* pp. 149–160 .

Tveit, A., 2001. *Peer-to-peer based recommendations for mobile commerceIn: Proceedings of the First International Mobile Commerce Workshop*, ACM Press, Rome, Italy pp. 26–29 .

Zhao, B. Y. et al (2001). *Tapestry: an infrastructure for fault-tolerant wide-area location and routing*. Tech. Rep. UCB/CSB-0-114, UC Berkeley, EECS.

Corresponding author. Tel.:/fax: +86-21-62933205

Expert Systems with Applications
Volume 27, Issue 2, August 2004, Pages 203-210

Expert Systems with Applications
Volume 27, Issue 2, August 2004, Pages 203-210

doi:10.1016/j.eswa.2004.01.003 Cite or Link Using DOI

A scalable P2P recommender system based on distributed collaborative filtering

Peng Han, Bo Xie, Fan Yang and Ruimin Shen
Department of Computer Science and Engineering, Shanghai Jiaotong University, 6th Floor Haoran High-tech Building, Shanghai 200030, China

Available online 10 February 2004.

Abstract

Collaborative Filtering (CF) technique has been proved to be one of the most successful techniques in recommender systems in recent years. However, most existing CF based recommender systems worked in a centralized way and suffered from its shortage in scalability as their calculation complexity increased quickly both in time and space when the record in user database increases. In this article, we first propose a distributed CF algorithm called PipeCF together with two novel approaches: significance refinement and unanimous amplification, to further improve the scalability and prediction accuracy. We then show how to implement this algorithm on a Peer-to-Peer (P2P) structure through distributed hash table method, which is the most popular and efficient P2P routing algorithm, to construct a scalable distributed recommender system. The experimental data show that the distributed CF-based recommender system has much better scalability than traditional centralized ones with comparable prediction efficiency and accuracy.

Author Keywords: Recommender system; Collaborative filtering; Peer-to-Peer; Significance refinement; Unanimous amplification

Article Outline

1. Introduction

Recommender system is a system that helps users to find their wanted items by making recommendations based on either the content of the recommended items (Content-based Filtering), or ratings of similar users on the recommended items (Collaborative Filtering, CF). Since Goldberg, Nichols, Oki, and Terry (1992) published the first account of using CF for information filtering; CF has proved to be one of the most successful techniques in recommendation systems by its advantage of that no explicit description of items is needed. The key idea of CF is that users will prefer those items that people with similar interests prefer, or even that dissimilar people do not prefer, so most CF algorithms can be separated into three steps as addressed by Herlocker, Konstan, Borchers, and Riedl (1999): (1) Similarity Weight: weight all users with respect to similarity with the active user, which refer to the user whose preferences are to be predicted; (2) Selecting Neighborhoods: select those users used to make prediction; (3) Rating Normalization and Prediction Making: normalize and calculate the weighted sum of selected users' ratings, then make prediction based on that. According to different techniques used in the first part mentioned above, CF algorithms can be divided into two classes: memory-based algorithms and model-based algorithms. Breese et al. performed an empirical analysis on both of two kinds of CF algorithms in Breese, Heckerman, and Kadie (1998) while Herlock et al. presented an algorithmic framework for performing CF in Herlocker et al. (1999).

GroupLens (Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994) was the first CF algorithm to automate prediction and used a memory-based algorithm. Like most memory-based algorithms, GroupLens need to

compute across the whole user database to calculate the similarities between active user and other users to make prediction. Ringo (Shardanand & Maes, 1995) only used those neighbors whose correlation were greater than a given threshold to make prediction. This approach not only reduced the calculation complexity but also proved to improve the performance. By choosing top-N users with the highest correlations the same improvement can also been obtained. However, all the other users' similarities still have to be calculated and its complexity increased quickly both in time and space as the record in the database increases.

Basically, there are two ways to reduce this calculation complexity. The first one is used a model-based algorithm which first constructs some certain mathematical models, such as Bayesian Network, Bayesian Classifiers et al., to describe the users and/or their ratings, then learns these models from the database and use them to make prediction. However, these approaches also need complex calculation when compiling models and also require a central database to keep all the user data which is not easy to achieve sometime not only for techniques reasons but also for privacy reasons.

The second way is to implement CF in a decentralized way. In fact, as Peer-to-Peer (P2P) gains more and more popularity, some researchers have already begun to consider it as an alternative architecture to reduce the calculation complexity (Tveit, 2001; Olsson, 2003 and Canny, 2002) of centralized CF algorithms. The main difference between centralized CF-based recommender system and distributed ones is that the originally centralized user database are maintained in a decentralized way which means each peer will only keep a fraction of user database and when making prediction for a particular user, needed record should first be retrieved to the user's own database from other peers and calculated locally. In order to do this, the following two problems have to be addressed: (1) how to store the user database distributed efficient so that needed information can be found efficiently; (2) how to identify those records needed to make prediction for a particular user and fetch them efficiently as retrieving all other users' votes back is not only unreasonable but also unnecessary.

The main contributions of this article are:

(1) We propose PipeCF: a distributed CF algorithm which can be implemented on a P2P overlay network;
(2) We propose two novel approaches: significance refinement (SR) and unanimous amplification (UA), to improve the performance of our distributed CF algorithm;
(1) We give the framework of implementing our distributed CF algorithm on P2P overlay network through distributed hash table (DHT) based technique to obtain efficient user database management and retrieval to construct decentralized CF recommender system.

The rest of this article is organized as follows. In Section 2, several related works are presented and discussed. In Section 3, we introduce the architecture and key features of our DHT-based CF system. Two techniques: SR and UA are also proposed in this section to improve the scalability and prediction accuracy of DHT-based CF algorithm in this section. In Section 4 the experimental results of our system are presented and analyzed. Finally we make a brief concluding remark and give the future work in Section 5.

2. Related work

2.1. Memory-based CF algorithm

Generally, the task of CF is to predict the votes of active users from the user database which consists of a set of votes $v_{i,j}$ corresponding to the vote of user *i* on item *j*. Memory-based CF algorithm calculates this prediction of as a weighted average of other users votes on that item through the following formula:

$$P_{a,j} = \bar{v}_a + \kappa \sum_{i=1}^{n} \varpi(a,j)(v_{i,j} - \bar{v}_i) \tag{1}$$

where $P_{a,j}$ denotes the prediction of the vote for active user *a* on item *j* and *n* is the number of users in user database. $\bar{v}_i$ is the mean vote for user *i* as:

$$\bar{v}_i = \frac{1}{|I_i|} \sum_{j \in I_i} v_{i,j} \tag{2}$$

where I_i is the set of items on which user *i* has voted. The weights $\varpi(a,j)$ reflect the similarity between active user and users in the user database. κ is a normalizing factor to make the absolute values of the weights sum to unity.

Most memory-based algorithms use Eq. (1) to make prediction and only distinguish between the ways they calculate the weights:

2.1.1. Pearson correlation coefficient

Pearson correlation coefficient was first introduced into collaborative filtering as a weighting method in the GroupLens project The correlation between user *a* and *i* is:

$$\varpi(a,i) = \frac{\sum_j (v_{a,j} - \bar{v}_a)(v_{i,j} - \bar{v}_i)}{\sqrt{\sum_j (v_{a,j} - \bar{v}_a)^2 \sum_j (v_{i,j} - \bar{v}_i)^2}} \tag{3}$$

where the summations is calculated over those items for which both users *a* and *i* have voted.

2.1.2. Vector similarity

The vector similarity was first used to measure the similarity between two documents. Each document was viewed as a vector of word frequency and their similarity was computed as the cosine of the angle between these two vectors. In Collaborative Filtering, we treat each user record as a document and their votes as frequency of items. So the weights can now be calculated as:

$$\varpi(a,i) = \sum_j \frac{v_{a,j}}{\sqrt{\sum_{k \in I_a} v_{a,k}^2}} \frac{v_{i,j}}{\sqrt{\sum_{k \in I_i} v_{i,k}^2}} \tag{4}$$

2.2. P2P system and DHT routing algorithm

The term 'Peer-to-Peer' refers to a class of systems and applications that employ distributed resources to perform a critical function in a decentralized manner. With the pervasive deployment of computers, P2P is increasingly receiving attention in research and more and more P2P systems have been deployed on the Internet. Some of the benefits of a P2P approach include: improving scalability by avoiding dependency on centralized points; eliminating the need for costly infrastructure by enabling direct communication among clients; and enabling resource aggregation. Among all these applications, three main classes of peer-to-peer applications have emerged: parallelizable, content and file management, and collaborative.

As the main purpose of P2P systems are to share resources among a group of computers called peers in a distributed way, efficient and robust routing algorithms for locating wanted resource is critical to the performance of P2P systems. Among these algorithms, distributed hash table (DHT) algorithm is one of.the most popular and

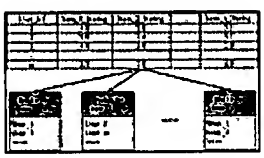
effective and supported by many P2P systems such as CAN (Ratnasamy, Francis, Handley, Karp, & Shenker, 2001), Chord (Stocal et al., 2001), Pastry (Rowstron & Druschel, 2001), and Tapestry (Zhao et al., 2001).

A DHT overlay network is composed of several DHT nodes and each node keeps a set of resources (e.g. files, rating of items). Each resource is associated with a key (produced, for instance, by hashing the file name) and each node in the system is responsible for storing a certain range of keys. Peers in the DHT overlay network locate their wanted resource by issue a *lookup(key)* request which returns the identity (e.g. the IP address) of the node that stores the resource with the certain key. The primary goals of DHT are to provide an efficient, scalable, and robust routing algorithm which aims at reducing the number of P2P hops, which are involved when we locate a certain resource, and to reduce the amount of routing state that should be preserved at each peer. In Chord (Stocal et al., 2001), each peer keeps track information of *logN* other peers (*N* is the total number of peers in the community). When a peer joins and leaves the overlay network, this highly optimized version of DHT algorithm will only require notifying *logN* peers about that change.

3. Our distributed CF algorithm

3.1. Basic PipeCF algorithm

The first step to implement CF algorithm in a distributed way is to divide the original centralized user database into fractions which can be stored in Peers distributed For concision, we use the term *bucket* to denote one fraction of the whole user database in the following of this article. Each bucket should also be assigned an identifier through which they can be located later when needed. The way we do the division is to make each bucket hold a group of users' record who has at least rated one item with the same vote. By that we construct one bucket for every different <ITEM_ID, VOTE> tuple and use that tuple as the identifier for the bucket. Fig. 1 shows our division strategy:



Display Full Size version of this image (4K)

Fig. 1. User database division strategy.

In order to reduce the calculation complexity and achieve scalability, we wish to find a strategy which can choose neighbors from the most suitable buckets when making prediction. The PipeCF algorithm chooses neighbors based on the heuristic that people with similar interests at least rate one item with similar votes. As we can see in Fig. 2, this strategy have very high hitting ratio. So when making prediction, the PipeCF only uses those users' records who are in at least one same bucket with the active users. Through which we reduce about 50% calculation than traditional CF algorithm and obtain comparable prediction as shown in Figure 8.

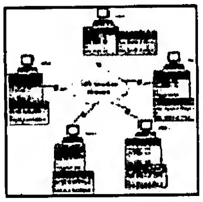

Display Full Size version of this image (8K)

Fig. 2. Architecture of DHT-based CF recommender system.

3.2. Improved PipeCF algorithm

3.2.1. Significance refinement

In the basic PipeCF algorithm, we use all users which are in the same bucket with the active user and find that the algorithm has an O(*N*) fetched user number (*N* is the total user number) as Fig. 8 shows. In fact, as Breese et al. (1998) presented by the term inverse user frequency, universally liked items are not as useful as less common items in capturing similarity. So we introduce a new concept SR, which reduces the returned user number of the basic PipeCF algorithm by limiting the number of returned users for each bucket. We term the algorithm improved by SR as Return K which means 'for every item, the PipeCF algorithm returns no more than K users for each bucket'. The experimental result in Section 5.3.3 shows that this method reduces the returned user number dramatically and also improves the prediction accuracy.

3.2.2. Unanimous amplification

Enlightened by the method of case amplification (Breese et al., 1998) which emphasizes the contribution of the most similar users to the prediction by amplifying the weights close to 1, we argue that we should give special award to the users who rated some items with the same vote by amplify their weights, which we term UA We transform the estimated weights as follows:

$$w'_{a,i} = \left\{ \begin{array}{ll} w_{a,i} & N_{a,i} = 0 \\ w_{a,i}\alpha & 0 < N_{a,i} \le \gamma \\ w_{a,i}\beta & N_{a,i} > \gamma \end{array} \right\} \tag{5}$$

where $N_{a,i}$ denotes the number of items which user *a* and user *i* have the same votes. A typical value for α for our experiments is 2.0, β is 4.0, and γ is 4. Experimental result in Section 4.3.4 shows that UA approach improves the prediction accuracy of the PipeCF algorithm.

4. DHT-BASED CF recommender system

4.1. Architecture of DHT-based CF recommender system

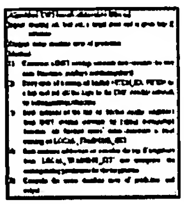
The main advantage of our DHT-based CF recommender system vs. traditional centralized CF recommender system is that both the maintenance of user database and the complex computation task of making prediction are done in a decentralized way so as to obtain better scalability. Here, we treat each bucket as resource and the unique key for the resource is generated by the particular <ITEM_ID, VOTE> tuple associated with the bucket. Each peer in the DHT overlay network will keep one or several buckets locally. Fig. 2 gives the architecture of our DHT-based CF recommender system.

So the implementation of PipeCF in DHT overlay network is straightforward except that the bucket is stored

distributed so that when a user wants to look up other similar users which have the same particular <ITEM_ID, VOTE> tuple, it need fetch them from DHT overlay network. Still we need special scheme to manage the distributed storage of buckets. DHT has provided two function *lookup(key)* and *put(key)* which we will describe later to do these jobs and its efficiency has been guaranteed by the algorithm itself. So with the DHT overlay network, all the users in the CF system are connected together and can find their wanted similar neighbors efficiently through a DHT routing algorithm.

4.2. Implementation of PipeCF on DHT

On the basis of the decentralized storage of user votes, we introduce our implementation of PipeCF algorithm on the DHT overlay network, called DHT-based CF algorithm, as shown in Fig. 3.



Display Full Size version of this image (17K)

Fig. 3. DHT-based CF algorithm.

There are two key pieces to the DHT-based CF system algorithm: the lookup mechanism used to locate similar users and fetch their actual rating. The decentralized storage (and hence decentralized retrieval) in decentralized CF system makes the CF calculation inherently scalable (every user do recommendation locally instead of depending on a centralized server); the hard part is finding the similar peers from which to retrieve the actual rating.

We devise a scalable solution to the problem of locating similar users in decentralized CF system, i.e. give a user vote vector; we can find the IP address of the node(s) which is similar to the user. Our DHT-based solution can reach following goal:

• *Scalability*: it must be designed to scale to several million nodes.
• *Efficiency*: similar users should be located reasonably quick and with low overhead in terms of the message traffic generated.
• *Dynamicity*: the system should be robust to frequent node arrivals and departures in order to cope with highly transient user populations' characteristic to decentralized environments.
• *Balanced load*: in keeping with the decentralized nature, the total resource load (traffic, storage, etc) should be roughly balanced across all the nodes in the system.

So we only select similar users in the subset in which users have same <ITEM_ID, VOTE> tuple. The key idea of our algorithm is hashing every user for every rated item. Our DHT-based CF algorithm includes two main DHT functions: *put(key)* and *lookup(key)*, and Fig. 4 and Fig. 5 show them separately.

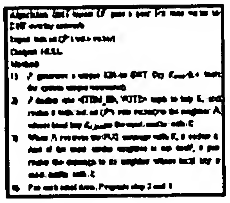
 **Display Full Size version of this image** (15K)

Fig. 4. DHT-based CF put function.

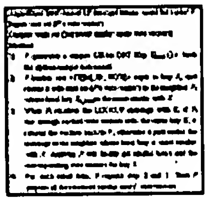
 **Display Full Size version of this image** (19K)

Fig. 5. DHT-based CF lookup function.

DHT-based CF Put algorithm is used to construct DHT overlay network and fill data in it. DHT-based CF Lookup algorithm is used to lookup and fetch similar uses with same <ITEM_ID, VOTE> tuple in order to construct a local training set to make recommendation. The main purpose of steps 2 and 3 in Fig. 3 is to make every peer in the DHT overlay network keep several buckets which contain a group of users with same <ITEM_ID, VOTE> tuple, from which the Lookup algorithm can fetch similar users later in its steps 2 and 3.

5. Experimental evaluation

In this section, we describe the dataset, metrics and methodology for the comparison between traditional and DHT-based CF algorithm, and present the results of our experiments.

5.1. Data set

We use Eachmovie collaborative filtering data set (1997) to evaluate the performance of improved algorithm The EachMovie data set is provided by the Compaq System Research Center, which ran the EachMovie recommendation service for 18 months to experiment with a collaborative filtering algorithm. The information they gathered during that period consists of 72,916 users, 1628 movies, and 2,811,983 numeric ratings ranging from 0 to 5. To speed up our experiments, we only use a subset of the EachMovie data set.

5.2. Metrics and methodology

The metrics for evaluating the accuracy of a prediction algorithm can be divided into two main categories: statistical accuracy metrics and decision-support metrics. Statistical accuracy metrics evaluate the accuracy of a predictor by comparing predicted values with user-provided values. Decision-support accuracy measures how well predictions help user select high-quality items. We use Mean Absolute Error (MAE), a statistical accuracy metrics, to report prediction experiments for it is most commonly used and easy to understand:

$$\mathrm{MAE} = \frac{\sum_{a \in T} |v_{a,j} - p_{a,j}|}{|T|} \qquad (6)$$

where $v_{a,j}$ is the rating given to item j by user a, is the predicted value of user a on item j, T is the test set, $|T|$ is the size of the test set.

We select 2000 users and choose one user as active user per time and the remainder users as his candidate neighbors, because every user only make self's recommendation locally. We use the mean prediction accuracy of all the 2000 users as the system's prediction accuracy. For every user's recommendation calculation, our tests are performed using 80% of the user's ratings for training, with the remainder for testing.

5.3. Experimental result

We design several experiments for evaluating our algorithm and analyze the effect of various factors (e.g. SR and UA, etc.) by comparison. All our experiments are run on a Windows 2000-based PC with Intel Pentium 4 processor having a speed of 1.8 GHz and 512 MB of RAM.

5.3.1. The efficiency of neighbor choosing

We used a data set of 5000 users and show among the users chosen by PipeCF algorithm, how many are in the top-100 users in Fig. 6. We can see from the data that when the user number rises above 1000, more than 80 users who have the most similarities with the active users are chosen by PipeCF algorithm.

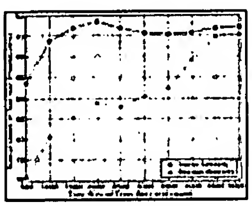

Display Full Size version of this image (5K)

Fig. 6. How many users of PipeCF in traditional CF's top 100.

5.3.2. Performance comparison

We compare the prediction accuracy of traditional CF algorithm and PipeCF algorithm while we apply both top-all and top-100 user selection on them. The results are shown as Fig. 7. We can see that the DHT-based algorithm has better prediction accuracy than the traditional CF algorithm.

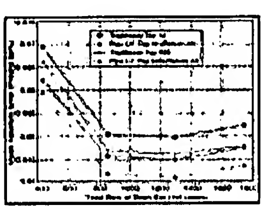

Display Full Size version of this image (5K)

Fig. 7. PipeCF vs. traditional CF.

5.3.3. The effect of significance refinement

We limit the number of returned user for each bucket by 2 and 5 and do the experiment in Section 5.3.2 again. The user for each bucket is chosen randomly. The result of the number of user chosen and the prediction accuracy is shown in Fig. 8 and Fig. 9, respectively. The result shows:

(1) 'Return All' has an *O(N)* returned user number and its prediction accuracy is also not satisfying;
(2) 'Return 2' has the least returned user number but the worst prediction accuracy;
(1) 'Return 5' has the best prediction accuracy and the scalability is still reasonably well (the returned user number is still limited to a constant as the total user number increases).

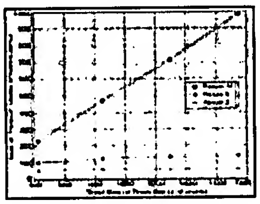

Display Full Size version of this image (5K)

Fig. 8. The effect on scalability of SR on PipeCF.

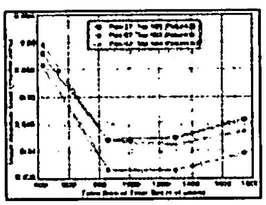

Display Full Size version of this image (5K)

Fig. 9. The effect on prediction accuracy of SR on PipeCF algorithm.

5.3.4. The effect of unanimous amplification

We adjust the weights for each user by using Eq. (5) while setting value for α as 2.0, β as 4.0, γ as 4 and do the experiment in Section 4.3.2 again. We use the top-100 and 'Return All' selection method. The result shows that the UA approach improves the prediction accuracy of both the traditional and the PipeCF algorithm. From Fig. 10 we can see that when UA approach is applied, the two kinds of algorithms have almost the same performance.

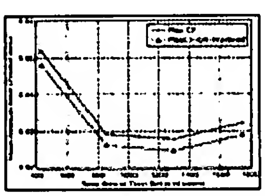

Display Full Size version of this image (4K)

Fig. 10. The effect on prediction accuracy of unanimous amplification.

6. Conclusion and future work

In this article, we propose a novel distributed hash table (DHT) based technique to implement efficient user

database management and retrieval in decentralized CF system. Then we propose a heuristic algorithm to fetch similar users from DHT overlay network and do recommendation locally. Finally, we propose two novel approaches: SR and UA to improve the performance of our DHT-based CF algorithm. The experimental data show that our DHT-based CF system has better prediction accuracy, efficiency and scalability than traditional CF systems.

Our future work includes investigation on a more efficient decentralized user database management and K-Nearest Neighbor (KNN) methods which can dynamically self-organize users with semantic similar interests combining content-based filtering techniques. We would also like to investigate on the influence of parameters choosing in UA.

Acknowledgements

The work described in this article is supported partially by National Natural Science Foundation of China under Grant No. 60372078.

References

Breese, J., Heckerman, D. and Kadie, C., 1998. Empirical analysis of predictive algorithms for collaborative filtering. *In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pp. 43–52.

Canny, J., 2002. *Collaborative filtering with privacyIn: Proceedings of the IEEE Symposium on Research in Security and Privacy, Oakland, CA*, IEEE Computer Society, Technical Committee on Security and Privacy, IEEE Computer Society Press pp. 45–57 .

Eachmovie collaborative filtering data set, (1997). http://research.compaq.com/SRC/eachmovie.

Goldberg, D., Nichols, D., Oki, B.M. and Terry, D., 1992. Using collaborative filtering to weave an information tapestry. *Communications of the ACM* **35** 12, pp. 61–70. **Full Text** via CrossRef

Herlocker, J.L., Konstan, J.A., Borchers, A. and Riedl, J., 1999. An algorithmic framework for performing collaborative filtering. *In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 230–237. **Full Text** via CrossRef

Olsson, T (2003). *Bootstrapping and Decentralizing Recommender Systems*. Licentiate Thesis 2003-006. Department of Information Technology, Uppsala University and SICS.

Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S., 2001. A scalable content-addressable network. *In: SIGCOMM.*

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J., 1994. *GroupLens: an open architecture for collaborative filtering of netnewsIn: Proceedings of the 1994 ACM conference on Computer supported cooperative work, October 22–26, 1994, Chapel Hill, North Carolina, United States* pp. 175–186 .

Rowstron, A. and Druschel, P., 2001. *Pastry: Scalable, distributed object location and routing for large scale peer-to-peer systemsIn: IFIP/ACM Middleware*, Hedelberg, Germany.

Shardanand, U. and Maes, P., 1995. *Social information filtering: algorithms for automating 'word of mouth'In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, May 07–11, 1995, Denver, Colorado, United States* pp. 210–217 .

Stocal, I. *et al.*, 2001. *Chord: a scalable peer-to-peer lookup service for Internet applicationsIn: ACM SIGCOMM, San Diego, CA, USA* pp. 149–160 .

Tveit, A., 2001. *Peer-to-peer based recommendations for mobile commerceIn: Proceedings of the First International Mobile Commerce Workshop*, ACM Press, Rome, Italy pp. 26–29 .

Zhao, B. Y. et al (2001). *Tapestry: an infrastructure for fault-tolerant wide-area location and routing.* Tech. Rep. UCB/CSB-0-114, UC Berkeley, EECS.

Corresponding author. Tel.:/fax: +86-21-62933205

Expert Systems with Applications
Volume 27, Issue 2, August 2004, Pages 203-210

Expert Systems with Applications
Volume 27, Issue 2, August 2004, Pages 203-210

doi:10.1016/j.eswa.2004.01.003

A scalable P2P recommender system based on distributed collaborative filtering

Peng Han, Bo Xie, Fan Yang and Ruimin Shen
Department of Computer Science and Engineering, Shanghai Jiaotong University, 6th Floor Haoran High-tech Building, Shanghai 200030, China

Available online 10 February 2004.

Abstract

Collaborative Filtering (CF) technique has been proved to be one of the most successful techniques in recommender systems in recent years. However, most existing CF based recommender systems worked in a centralized way and suffered from its shortage in scalability as their calculation complexity increased quickly both in time and space when the record in user database increases. In this article, we first propose a distributed CF algorithm called PipeCF together with two novel approaches: significance refinement and unanimous amplification, to further improve the scalability and prediction accuracy. We then show how to implement this algorithm on a Peer-to-Peer (P2P) structure through distributed hash table method, which is the most popular and efficient P2P routing algorithm, to construct a scalable distributed recommender system. The experimental data show that the distributed CF-based recommender system has much better scalability than traditional centralized ones with comparable prediction efficiency and accuracy.

Author Keywords: Recommender system; Collaborative filtering; Peer-to-Peer; Significance refinement; Unanimous amplification

Article Outline

Expert Systems with Applications
Volume 27, Issue 2, August 2004, Pages 203-210

doi:10.1016/j.eswa.2004.01.003 Cite or Link Using DOI

A scalable P2P recommender system based on distributed collaborative filtering

Peng Han, Bo Xie, Fan Yang and Ruimin Shen
Department of Computer Science and Engineering, Shanghai Jiaotong University, 6th Floor Haoran High-tech Building, Shanghai 200030, China

Available online 10 February 2004.

Abstract

Collaborative Filtering (CF) technique has been proved to be one of the most successful techniques in recommender systems in recent years. However, most existing CF based recommender systems worked in a centralized way and suffered from its shortage in scalability as their calculation complexity increased quickly both in time and space when the record in user database increases. In this article, we first propose a distributed CF algorithm called PipeCF together with two novel approaches: significance refinement and unanimous amplification, to further improve the scalability and prediction accuracy. We then show how to implement this algorithm on a Peer-to-Peer (P2P) structure through distributed hash table method, which is the most popular and efficient P2P routing algorithm, to construct a scalable distributed recommender system. The experimental data show that the distributed CF-based recommender system has much better scalability than traditional centralized ones with comparable prediction efficiency and accuracy.

Author Keywords: Recommender system; Collaborative filtering; Peer-to-Peer; Significance refinement; Unanimous amplification

Article Outline